

48

41608

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	:	
HÖLLIG ET AL.	:	PATENT
Serial No.: 09/853,026	:	Group Art Unit: 2123
Filed: May 11, 2001	:	Examiner:
For: PROCESS FOR INCREASING THE	:	
EFFICIENCY OF A COMPUTER IN	:	
FINITE ELEMENT SIMULATIONS AND	:	
A COMPUTER FOR PERFORMING	:	
THAT PROCESS	:	

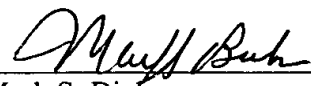
SUPPLEMENT TO CLAIM FOR PRIORITY UNDER 35 U.S.C. § 119

Commissioner for Patents
Washington, D.C. 20231

Sir:

Filed herewith is a verified translation of the priority application.

Respectfully submitted,



Mark S. Bicks
Reg. No. 28,770

ROYLANCE, ABRAMS, BERDO &
GOODMAN, L.L.P.
1300 19th Street, N.W., Suite 600
Washington, D.C. 20036
(202) 659-9076

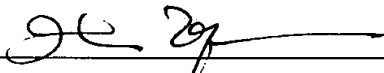
Dated: Oct 22, 2001

DECLARATION OF TRANSLATOR

I, Joachim Wipper, of Stuttgart University,
do hereby avow and declare that I am conversant with the English and German languages and am
a competent translator of German into English. I declare further that to the best of my
knowledge and belief the following are true and correct translations prepared and reviewed by
me of the documents in the German language attached hereto.

I hereby declare that all statements made herein of my own knowledge are true and that
all statements made on information and belief are believed to be true; and further that these
statements were made with the knowledge that willful false statements and the like so made are
punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States
Code and that such willful false statements may jeopardize the validity of the attached patent
application or any patent issued thereon.

Date: 09/18/2001

Name: 
Joachim Wipper

DECLARATION OF TRANSLATOR

I, ULRICH REIF, of Sandweg 44, 64385 Reichelsheim, Germany,
do hereby avow and declare that I am conversant with the English and German languages and am
a competent translator of German into English. I declare further that to the best of my
knowledge and belief the following are true and correct translations prepared and reviewed by
me of the documents in the German language attached hereto.

I hereby declare that all statements made herein of my own knowledge are true and that
all statements made on information and belief are believed to be true; and further that these
statements were made with the knowledge that willful false statements and the like so made are
punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States
Code and that such willful false statements may jeopardize the validity of the attached patent
application or any patent issued thereon.

Date: 09/20/2006

Ulrich Reif
Name:

DECLARATION OF TRANSLATOR

I, Klaus Höllig, of Univ. of Stuttgart,

do hereby avow and declare that I am conversant with the English and German languages and am a competent translator of German into English. I declare further that to the best of my knowledge and belief the following are true and correct translations prepared and reviewed by me (*) of the documents in the German language attached hereto.

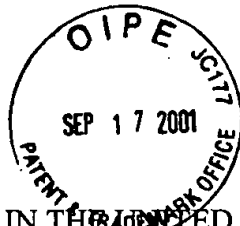
I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the attached patent application or any patent issued thereon.

Date: 09/10/2001

Klaus Höllig
Name:

(*) jointly with U. Reif & J. Wipper

41608



#7

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of :
HÖLLIG ET AL. : PATENT
Serial No.: 09/853,026 : Group Art Unit: 2123
Filed: May 11, 2001 : Examiner:
For: PROCESS FOR INCREASING THE :
EFFICIENCY OF A COMPUTER IN :
FINITE ELEMENT SIMULATIONS AND :
A COMPUTER FOR PERFORMING :
THAT PROCESS :

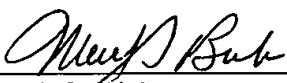
CLAIM FOR PRIORITY UNDER 35 U.S.C. § 119

Commissioner for Patents
Washington, D.C. 20231

Sir:

Under the provisions of 35 U.S.C. § 119, filed herewith is a certified copy of German Application No. 100 23 377.5, filed 12 May 2000, in accordance with the International Convention for the Protection of Industrial Property, 53 Stat. 1748, under which Applicants hereby claim priority, and a verified translation of the priority application.

Respectfully submitted,


Mark S. Bicks
Reg. No. 28,770

ROYLANCE, ABRAMS, BERDO &
GOODMAN, L.L.P.
1300 19th Street, N.W., Suite 600
Washington, D.C. 20036
(202) 659-9076

Dated: Sept 14, 2001



**Prioritätsbescheinigung über die Einreichung
einer Patentanmeldung**

Aktenzeichen: 100 23 377.5

Anmeldetag: 12. Mai 2000

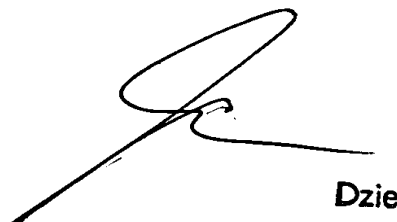
Anmelder/Inhaber: Universität Stuttgart,
Stuttgart/DE

Bezeichnung: Verfahren zur Erhöhung der Leistungs-
fähigkeit einer Computereinrichtung bei
Finite-Elemente-Simulationen und eine
solche Computereinrichtung

IPC: G 06 F, G 06 T

**Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ur-
sprünglichen Unterlagen dieser Patentanmeldung.**

München, den 10. Mai 2001
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag



Dzierzon

Stuttgart, den 11. Mai 2000

Universität Stuttgart
Keplerstraße 7, D-70174 Stuttgart

Verfahren zur Erhöhung der Leistungsfähigkeit
einer Computereinrichtung bei Finite-Elemente-Simulationen
und eine solche Computereinrichtung

Die Erfindung betrifft ein Verfahren zur Erhöhung der Leistungsfähigkeit einer Computereinrichtung bei Finite-Elemente-Simulationen durch effiziente automatische Konstruktion geeigneter Basisfunktionen für die Berechnung von Näherungslösungen und eine solche Computereinrichtung.

5

Eine Vielzahl technisch-physikalischer Phänomene lässt sich durch partielle Differenzialgleichungen beschreiben. Hierunter fallen u.a. Probleme aus der Strömungsmechanik (z.B. Strömung um eine Tragfläche), der elektromagnetischen Feldtheorie (z.B. elektrischer Feldverlauf in einem Transistor) oder der

- 10 Elastizitätstheorie (z.B. Deformation einer Autokarosserie). Die genaue Kenntnis und Beschreibung derartiger Vorgänge ist ein zentraler Bestandteil bei der Konstruktion und Optimierung technischer Objekte. Um zeit- und kostenintensive Experimente einzusparen ist das Interesse an computergestützten Simulationen groß. Hierbei haben sich Finite-Elemente-Verfahren (FE-Verfahren) etabliert und
- 15 sind seit langem Gegenstand intensiver Forschung. Dies betrifft insbesondere auch automatische Netzgenerierungsverfahren als Grundlage für die Konstruktion geeigneter Basisfunktionen.

Fig. 1 veranschaulicht in der linken Bildhälfte den Stand der Technik beim Ablauf einer FE-Simulation für ein lineares Randwertproblem als typisches Modellbeispiel. Ausgehend von den Daten, die die Geometrie des zu simulierenden technischen Objektes beschreiben, wird zunächst ein System von Basisfunktionen konstruiert, das einerseits die Einhaltung der Randbedingungen ermöglicht und andererseits zur Approximation der gesuchten Lösung geeignet ist. Dann wird mit Hilfe dieser Basisfunktionen ein lineares Gleichungssystem mittels numerischer Integrationsverfahren aufgestellt. Schließlich werden die Koeffizienten der gesuchten Approximation als Lösung dieses linearen Gleichungssystems ermittelt.

10

FE-Verfahren oder deren Anwendung sind Gegenstand einer Reihe von Patenten. Beispielhaft sei US 4,819,161 genannt, wo die FE-Approximationen einer großen Klasse von Differenzialgleichungen automatisiert werden, sowie US 5,731,817, wo ein Verfahren zur Generierung von Hexaeder-Netzen die Grundlage für ein Verfahren zur FE-Simulation bildet.

15

Bei den meisten praxisrelevanten FE-Verfahren werden Basisfunktionen verwendet, die auf einer durch Vernetzung des Simulationsgebietes erzeugten Zerlegung definiert sind. Fig. 2a zeigt eine Auswahl gebräuchlicher Elemente deren

20

Dimension, Grad, Glattheit und Parameter in Fig. 2b aufgelistet sind. Eine Übersicht über Vernetzungsmethoden ebener Gebiete findet sich beispielsweise in K. Ho-Le, Finite element mesh generation methods: a review and classification, Comp. Aided Design 20 (1988), 27-38. Die Vernetzung komplizierterer dreidimensionaler Gebiete gestaltet sich nach derzeitigem Kenntnisstand überaus schwierig, wie S. Owen, A survey of unstructured mesh generation technology, Proceedings, 7th International Meshing Round Table, Sandia National Lab (1998), 239-257, zeigt. Die Verfahren benötigen lange Rechenzeiten und sind teilweise auch noch nicht vollkommen automatisiert. Es gibt jedoch in jüngerer Zeit eine Reihe sehr innovativer neuer Ansätze. Beispielsweise wird in A. Fuchs, Optimierte Delaunay-Triangulierungen zur Vernetzung getrimmter NURBS-Körper,

30

Universität Stuttgart, 1999 eine Kräfteverteilung simuliert, um eine optimale Verteilung von Triangulierungspunkten zu erzielen. In US 5,729,670 werden zwei- und dreidimensionale Netze durch Lösen von Strömungsproblemen erzeugt, was eine interessante Umkehrung des üblichen FE-Mechanismusses ist. Daneben

- 5 wurden viele Algorithmen entwickelt, um einzelne Teilaspekte der Netzgenerierung zu verbessern. Beispielsweise wird in DE 196 21 434 A1 und US 5,774,696 ein Verfahren zur Elimination von Überschneidungen mit vorgegebenen Kanten oder Grenzflächen bei Delaunay-Triangulierungen beschrieben.

- 10 Vernetzungsfreie FE-Verfahren haben bisher keine Bedeutung für die Anwendungen erlangt. Sowohl bei dem Lagrange-Multiplier-Verfahren, s. z.B. J.H. Bramble, The Lagrange multiplier method for Dirichlet's problem, Math. Comp. 37 (1981), 1-11, als auch bei der Penalty-Methode, s. z.B. P. Bochev und M. Gunzburger, Finite element methods of least squares type, SIAM Review 40
15 (1998), 789-837, stellt bei Verwendung einfacher, stabiler Basisfunktionen die Behandlung von Randbedingungen ein Hauptproblem dar.

Bei vielen technischen Simulationen ist die automatische Netzgenerierung sehr aufwendig und nimmt den weitaus größten Teil der Rechenzeit in Anspruch. Des

- 20 Weiteren ist die Approximationskraft der üblicherweise verwendeten linearen und multilineareren Basisfunktionen gering. Um genaue Resultate zu erzielen, muss also eine große Anzahl von Basisfunktionen verwendet und damit ein entsprechend großes Gleichungssystem gelöst werden. Ansatzfunktionen höheren Grades über Triangulierungen besitzen in der Regel ebenfalls ein ungünstiges Verhältnis
25 zwischen der erreichbaren Genauigkeit und der Anzahl der verwendeten Basisfunktionen. Schließlich lassen sich auf unstrukturierten Netzen keine glatten Basisfunktionen auf einfache Weise definieren. Bereits für stetig differenzierbare Elemente sind sehr spezielle Konstruktionen notwendig (s. Fig. 2a).

Der Erfindung liegt das Problem zugrunde, die Leistungsfähigkeit bekannter FE-Verfahren und die FE-Verfahren ausführender Computereinrichtungen durch die effiziente Konstruktion von Basisfunktionen mit günstigen Eigenschaften zu erhöhen. Insbesondere soll die Vernetzung des Simulationsgebietes vollständig
5 entfallen, eventuell gegebene Randbedingungen eingehalten werden, genaue Lösungen mit relativ wenigen Koeffizienten erreichbar und das entstehende Gleichungssystem effizient lösbar sein. Dadurch sollen die Nachteile des Standes der Technik überwunden und somit die Genauigkeit und Geschwindigkeit der Simulation physikalischer Eigenschaften bei der Konstruktion und Optimierung
10 technischer Objekte verbessert werden.

Einige zentrale Begriffe und Bezeichnungen, die in der nachfolgenden Beschreibung des erfindungsgemäßen Verfahrens verwendet werden, sollen vorab erläutert werden.

15

Das *Simulationsgebiet* Ω ist eine beschränkte Menge der Dimension $d = 2$ oder $d = 3$, auf der die zu untersuchenden physikalischen Größen mittels der FE-Methode approximiert werden sollen. Der *Rand* des Simulationsgebietes wird mit Γ bezeichnet. Unter einem *Gitter* mit *Gitterweite* h versteht man eine
20 Zerlegung einer Teilmenge der Ebene oder des Raumes in *Gitterzellen* Z_k . Dabei ist jede Gitterzelle je nach Dimension d ein Quadrat oder ein Würfel mit Kantenlänge h . Genauer gilt $Z_k = kh + [0, h]^d$, wobei k eine Menge von ganzzahligen d -Vektoren durchläuft. Die *uniformen Tensorprodukt B-Splines* in d Variablen vom *Grad* n zur *Gitterweite* h werden mit b_k bezeichnet, s. z.B. C. de
25 Boor, A Practical Guide to Splines, Springer, 1978. Dies sind $(n - 1)$ -mal stetig differenzierbare Funktionen, die auf den Gitterzellen mit Polynomen vom Grad n übereinstimmen, wie in Fig. 4 dargestellt. Die obere Bildhälfte zeigt einen *Träger* Q_k des B-Splines vom Grad $n = 2$, der Dimension $d = 2$ und der Glattheit $m = 1$. In der unteren Bildhälfte ist der resultierende Tensorprodukt B-Spline b_k
30 abgebildet. Der Träger Q_k , also die Vereinigung aller Gitterzellen, auf denen der

B-Spline b_k nicht identisch Null ist, besteht aus $(n + 1)^d$ Gitterzellen; es gilt genauer $Q_k = kh + [0, (n + 1)h]^d$. In allen Figuren ist der B-Spline b_k an der Stelle kh markiert, also beispielsweise im Fall $d = 2$ in der linken unteren Ecke des Trägers. Für die FE-Simulation sind nur solche B-Splines von Belang, deren Träger
5 das Simulationsgebiet Ω schneiden; sie werden als *relevante B-Splines* bezeichnet. Die relevanten B-Splines werden nochmals in zwei Gruppen eingeteilt: diejenigen B-Splines, bei denen der Teil des Trägers im Inneren des Simulationsgebietes größer als eine vorgegebene *Schranke* s ist, werden als *innere B-Splines* bezeichnet. Alle übrigen relevanten B-Splines werden als *äußere B-Splines* bezeichnet.

10

Die Aufgabe der vorliegenden Erfindung wird durch das im Anspruch 1 bestimmte Verfahren gelöst. Besondere Ausführungsarten der Erfindung sind in den Unteransprüchen bestimmt. Der Anspruch 11 bestimmt eine erfindungsgemäße Computereinrichtung.

15

In der rechten Bildhälfte der Fig. 1 ist die Einbindung des erfindungsgemäßen Verfahrens in den Ablauf einer FE-Simulation nach dem Stand der Technik bzw. die Substitution einiger Verfahrensschritte einer FE-Simulation nach dem Stand der Technik durch das erfindungsgemäße Verfahren dargestellt.

20

Die Eingabe 1 des Simulationsgebietes Ω kann über ein Eingabemittel erfolgen, insbesondere auch durch Einspeichern von Daten, die aus einer computerunterstützten Konstruktion (CAD/CAM) abgeleitet sind. Beispielsweise können so die bei der Konstruktion eines Kraftfahrzeugs verwendeten Daten
25 unmittelbar in die erfindungsgemäße FE-Simulation einfließen.

Bei der Eingabe 2 und Abspeicherung des Typs der Randbedingungen wird zwischen natürlichen und wesentlichen Randbedingungen unterschieden. Die erfindungsgemäße Basis wird für homogene Randbedingungen gleichen Typs
30 konstruiert. Insbesondere verschwinden bei wesentlichen Randbedingungen die

Basisfunktionen auf dem Rand Γ . Inhomogene Randbedingungen lassen sich dann beim Aufstellen des FE-Systems mittels Methoden, die dem Stand der Technik entsprechen, behandeln.

- 5 Schließlich werden Steuerparameter eingelesen 3. Diese betreffen den Grad n und die Gitterweite h der zu verwendenden B-Splines sowie die Schranke s zur Klassifikation der inneren und äußeren B-Splines. Bei fehlender Spezifikation können alle diese Eingabeparameter mit Hilfe von empirisch oder analytisch konstruierten Bewertungsfunktionen automatisch bestimmt werden.

10

Die folgende erfindungsgemäße Konstruktion der Basisfunktionen gliedert sich in die in Fig. 3 schematisch dargestellten Schritte, die nun beschrieben werden...

- Nach dem Einlesen des Simulationsgebietes Ω wird im ersten Verfahrensschritt ein
15 das Simulationsgebiet Ω überdeckendes Gitter erstellt. Dann wird geprüft, welche der Gitterzellen ganz, teilweise oder nicht im Simulationsgebiet Ω liegen, die Zelltypen bestimmt 4 und diese Information über die Zelltypen gespeichert. Hierzu sind im Wesentlichen Innen/Außen-Tests sowie Schnittbestimmungen zwischen dem Rand Γ des Simulationsgebietes Ω und den Strecken bzw. Quadraten, die die
20 Gitterzellen beranden, durchzuführen. Fig. 7 zeigt die Ein- und Ausgabedaten für diesen Verfahrensschritt.

- Im zweiten Verfahrensschritt werden zunächst anhand der Information über die Zelltypen die relevanten B-Splines bestimmt. Dann wird die Klassifikation 5 in
25 innere und äußere B-Splines durchgeführt; die entsprechenden Listen der Indizes werden mit I und J bezeichnet. Hierzu ist mit Hilfe der im ersten Verfahrensschritt gewonnenen Daten die Größe derjenigen Teile der Träger der B-Splines zu bestimmen, die im Inneren des Simulationsgebietes liegen und mit der vorgegebenen Schranke s zu vergleichen. Fig. 9 zeigt die Ein- und Ausgabedaten
30 für diesen Verfahrensschritt.

Im dritten Verfahrensschritt werden Kopplungskoeffizienten $e_{i,j}$ berechnet 6, die innere und äußere B-Splines gemäß der Vorschrift

$$B_i(x) = b_i(x) + \sum_{j \in J(i)} e_{i,j} b_j(x), \quad i \in I \quad (1)$$

- 5 verbinden. Jedem inneren B-Spline b_i wird damit ein erweiterter B-Spline B_i zugeordnet. Die Konstruktion und die Eigenschaften der Indexmengen $J(i)$ und der Kopplungskoeffizienten $e_{i,j}$ sind wie folgt gegeben. Die Indexmengen $J(i)$ bestehen aus Indizes äußerer B-Splines. Ihnen entsprechen komplementäre Indexmengen $I(j)$ von Indizes innerer B-Splines; es gehört also i genau dann zu $I(j)$, wenn j zu $J(i)$ gehört. Für einen gegebenen äußeren Index j ist die Indexmenge $I(j)$ ein Feld, also ein quadratischer bzw. würfelförmiger Bereich von $(n+1)^d$ inneren Indizes, der durch einen minimalen Abstand zum Index j gekennzeichnet ist. Für einen gegebenen äußeren Index j und einen inneren Index i in der Indexmenge $I(j)$ sei p_i dasjenige d -variate Polynom vom Grad n in jeder Variablen, das an der Stelle i den Wert 1 und an allen anderen Punkten des Feldes $I(j)$ den Wert 0 hat. Dann ist der Kopplungskoeffizient $e_{i,j}$ als Wert von p_i an der Stelle j gegeben, es gilt also $e_{i,j} = p_i(j)$. Die konkreten Werte der Kopplungskoeffizienten können entweder für verschiedene Grade und relative Positionen von j und $I(j)$ tabelliert werden oder aber mit Hilfe von Lagrangepolynomen einfach berechnet werden. Fig. 11 zeigt die Ein- und Ausgabedaten für diesen Verfahrensschritt.

- Sind natürliche Randbedingungen gegeben, so werden die in Gleichung (1) definierten erweiterten Splines ohne Modifikation zur weiteren Durchführung des FE-Verfahrens verwendet. Sind dagegen wesentliche Randbedingungen gegeben, so ist noch eine Gewichtung gemäß der Vorschrift

$$B_i(x) \leftarrow \frac{w(x)}{w(x_i)} B_i(x), \quad i \in I \quad (2)$$

durchzuführen. Die zugehörige Abfrage erfolgt in einem optionalen Verfahrensschritt 6a. Die so definierten Funktionen werden als gewichtete

erweiterte B-Splines (WEB-Splines) bezeichnet. Formal gesehen entsprechen die bei natürlichen Randbedingungen verwendeten erweiterten B-Splines dem Sonderfall $w(x) = 1$. Sie werden deshalb ebenfalls als WEB-Splines bezeichnet. Für den Fall wesentlicher Randbedingungen ist die Gewichtsfunktion w wie folgt

5 gekennzeichnet: Für alle Punkte x des Simulationsgebietes lässt sich $w(x)$ nach oben und unten durch von x unabhängige positive Konstanten mal den Abstand $\text{dist}(x)$ des Punktes x vom Rand Γ abschätzen. Mit anderen Worten ist w im Inneren von Ω positiv und verschwindet in der Nähe des Randes Γ so schnell wie die Abstandsfunktion dist . Für Simulationsgebiete, die durch elementare

10 geometrische Objekte (Kreise, Ebenen, Ellipsen, etc.) berandet sind, kann eine geeignete Gewichtsfunktion gegebenenfalls in expliziter analytischer Form angegeben werden. Anderenfalls sind Berechnungsvorschriften heranzuziehen, die typischerweise eine Glättung der Abstandsfunktion darstellen. Der Skalierungsfaktor $1/w(x_i)$ entsteht durch Auswertung der Gewichtsfunktion im

15 Gewichtspunkt x_i . Dies kann ein beliebiger Punkt im Träger des B-Splines b_i sein, der vom Rand mindestens den halben Schrankenwert $s/2$ entfernt ist.

Als Resultat des erfindungsgemäßen Verfahrens erhält man eine Berechnungsvorschrift für die WEB-Splines B_i (vgl. Definitionen (1) und (2)), die

20 alle günstigen Eigenschaften gemäß der Aufgabenstellung aufweisen. Damit kann nun das FE-Verfahren gemäß dem Stand der Technik weitergeführt werden. Es ist dabei jedoch möglich, nach Anspruch 11 die reguläre Gitterstruktur der Basisfunktionen vorteilhaft auszunutzen.

25 Zusammenfassend ist zu sagen, dass für das erfindungsgemäße Verfahren unter anderem die Kopplung der äußeren an die inneren B-Splines von Bedeutung ist. Diese bewirkt, dass die konstruierte Basis die für FE-Berechnungen wesentlichen Eigenschaften besitzt. Insbesondere ist eine erfindungsgemäße Basis B_i (i aus der Indexmenge I) gleichmäßig in Bezug auf die Gitterweite h stabil und besitzt die

30 gleiche Fehlerordnung wie die B-Splines b_k bei Approximation glatter Funktionen,

die die gleichen Randbedingungen erfüllen. Zum anderen ist durch die Verwendung der Gewichtsfunktion w die Einhaltung wesentlicher Randbedingungen gewährleistet.

- 5 Die zuvor beschriebenen Verfahrensschritte und deren Eigenschaften sowie die bevorzugte Ausführungsart und die zugehörigen Ausführungsbeispiele sollen mit Hilfe der nachfolgend aufgelisteten Figuren weiter erläutert bzw. illustriert werden. Dabei können die in den Ansprüchen und in der Beschreibung erwähnten Merkmale jeweils einzeln für sich oder in beliebiger Kombination
- 10 erfindungswesentlich sein.

Fig. 1 zeigt zum Stand der Technik die einzelnen Schritte beim Ablauf einer Finite-Elemente-Simulation und ordnet die erfindungsgemäße Ermittlung der WEB-Basis in diesen Ablauf ein,

15 Fig. 2a stellt einige Finite Elemente des Standes der Technik dem in Fig. 2b gezeigten WEB-Element gegenüber und listet die für Finite-Elemente-Approximationen relevanten Parameter auf,

Fig. 3 zeigt in einem Flussdiagramm die Verfahrensschritte zur Ermittlung der WEB-Basis,

20 Fig. 4 zeigt einen Träger und den zugehörigen Tensorprodukt-B-Spline vom Grad 2,

Fig. 5 illustriert die Problemstellung des ersten Ausführungsbeispiels (Auslenkung einer Membran unter konstantem Druck) und zeigt die zugehörige Lösung,

Fig. 6 zeigt die Zelltypen anhand des ersten Ausführungsbeispiels,

25 Fig. 7 gibt eine Übersicht der Ein- und Ausgabedaten des Verfahrens zur Bestimmung der Zelltypen,

Fig. 8 illustriert exemplarisch die Klassifikation der B-Splines anhand des ersten Ausführungsbeispiels,

- Fig. 9 gibt eine Übersicht der Ein- und Ausgabedaten des Verfahrens zur Klassifikation der B-Splines,
- Fig. 10 zeigt die Kopplungskoeffizienten eines äußeren B-Splines und die zugehörigen inneren B-Splines anhand des ersten Ausführungsbeispiels,
- Fig. 11 gibt eine Übersicht der Ein- und Ausgabedaten des Verfahrens zur Berechnung der Kopplungskoeffizienten,
- Fig. 12 illustriert die Konstruktion der Gewichtsfunktion der bevorzugten Ausführungsart,
- Fig. 13 zeigt den Träger eines WEB-Splines und die zugehörigen Kopplungskoeffizienten anhand des ersten Ausführungsbeispiels,
- Fig. 14 erläutert die Problemstellung eines zweiten Ausführungsbeispiels (inkompressible Strömung) und deren Lösung anhand der Strömungslinien und einer Abbildung zur Verteilung der Strömungsgeschwindigkeit,
- Fig. 15a bis 15c zeigen für das zweite Ausführungsbeispiel zu den B-Spline-Graden $n = 1, 2, 3$ und gleicher Gitterweite die sich ändernde Klassifikation der B-Splines,
- Fig. 16a bis 16c geben Auskunft über die Fehlerentwicklung bei der Finite-Elemente-Approximation mit Hilfe von WEB-Splines sowie über das Laufzeitverhalten der WEB-Approximation für das zweite Ausführungsbeispiel,
- Fig. 17 stellt die WEB-Basis einem auf linearen Ansatzfunktionen über einer Triangulierung (Stand der Technik) basierenden Verfahren gegenüber und
- Fig. 18 zeigt eine erfindungsgemäße Computereinrichtung.

Eine besonders günstige Ausführungsart des erfindungsgemäßen Verfahrens, die als WEB-Verfahren bezeichnet wird, ist durch folgende Festlegungen gegeben:

Die Schranke s wird so gewählt, dass die inneren B-Splines dadurch gekennzeichnet sind, dass mindestens eine der Gitterzellen ihres Trägers vollständig im Simulationsgebiet Ω liegt. Da zur Bestimmung der relevanten B-Splines ohnehin der Schnitt zwischen den Gitterzellen und dem Rand Γ zu berechnen ist, ist somit zur Klassifizierung kein wesentlicher zusätzlicher Rechenaufwand notwendig. Der Gewichtspunkt x_i wird als Mittelpunkt einer vollständig im Simulationsgebiet Ω liegenden Gitterzelle im Träger des B-Splines b_i gewählt. Auch dies ist effizient möglich, da die Bestimmung einer solchen Zelle bereits Bestandteil der Klassifizierungsroutine ist.

10

Falls keine explizite analytische Form der Gewichtsfunktion bekannt ist, wird diese wie folgt angesetzt:

$$w(x) = \begin{cases} 1 & \text{falls } \text{dist}(x) \geq \delta \\ 1 - (1 - \text{dist}(x)/\delta)^n & \text{falls } \text{dist}(x) < \delta \end{cases} \quad (3)$$

Fig. 12 illustriert die Konstruktion der Gewichtsfunktion. Dabei gibt der Parameter δ die Breite eines Streifens Ω_δ an, innerhalb dessen die Gewichtsfunktion zwischen dem Wert 0 auf dem Rand des Simulationsgebietes und dem Wert 1 auf einem Plateau über $\Omega \setminus \Omega_\delta$ variiert. Der Parameter δ wird so bestimmt, dass die Glattheit der Gewichtsfunktion gewährleistet ist.

Ein wesentlicher Vorteil des Verfahrens ist, dass keine Vernetzung des Simulationsgebietes erforderlich ist. Dies bewirkt in technischen Anwendungen eine deutliche Ersparnis an Rechenzeit und Speicherkapazität und vereinfacht den Simulationsablauf. Dabei ist die Verfahrensstruktur für zwei- und dreidimensionale Probleme formal und auch in der technischen Durchführung weitgehend identisch. Dies ermöglicht die zeit- und kostensparende Implementierung von Lösern für verschiedenste Anwendungen basierend auf einheitlichen Programmstrukturen. Die Verwendung von B-Splines entspricht dem industriellen Standard bei der Modellierung geometrischer Objekte und schafft damit eine natürliche Verbindung zwischen FE- und CAD/CAM-Anwendungen. Bestehende umfangreiche

Programmbibliotheken aus beiden Bereichen können zur Implementierung einer auf dem erfindungsgemäßen Verfahren beruhender FE-Simulation eingesetzt werden. Die nach dem WEB-Verfahren konstruierten Basisfunktionen besitzen alle Standardeigenschaften Finiten Elemente. Hierzu gehört insbesondere die Stabilität
5 der Basis. Diese bewirkt, dass beispielsweise bei linearen elliptischen Randwertproblemen die Konditionszahl des entstehenden Gleichungssystems bei kleiner werdender Gitterweite nicht schneller anwächst als bei optimalen Triangulierungen. Für die Anwendungen bedeutet dies beispielsweise, dass sich lineare Gleichungssysteme, wie sie bei der FE-Methode typischerweise entstehen,
10 mittels iterativer Verfahren effizient lösen lassen. Weiterhin ist bei gegebenem Grad die Approximationsordnung maximal und die Anzahl der benötigten Parameter minimal. Damit sind sehr genaue Approximationen schon mit relativ geringer Parameterzahl möglich. Konkret kann dies bedeuten, dass Genauigkeiten, die bislang den Einsatz von Großrechnern erforderten, auf Arbeitsplatzrechnern
15 erzielt werden können. Die reguläre Gitterstruktur der erfindungsgemäßen Basis ermöglicht eine sehr effiziente Implementierung, insbesondere bei der Aufstellung und Lösung von FE-Systemen. Ferner gelingt es durch Verwendung der Gewichtsfunktion Randbedingungen bei der Simulation einzuhalten, ohne die reguläre Gitterstruktur der Basisfunktionen zu beeinträchtigen. Schließlich kann
20 bei der Lösung linearer elliptischer Randwertprobleme durch den Einsatz von Mehrgitter-Verfahren zur Lösung des entstehenden linearen Gleichungssystems erreicht werden, dass die Gesamtlösungszeit proportional zur Anzahl der gesuchten Koeffizienten und damit optimal ist.

25 Das erfindungsgemäße Verfahren in der speziellen bevorzugten Ausführungsart (WEB-Verfahren) wird anhand des in Fig. 5 gezeigten ersten Ausführungsbeispiels veranschaulicht. Die Differenzialgleichung und Randbedingungen sind sehr elementar gewählt, so dass ohne großen Mehraufwand neben der erfindungsgemäßen Konstruktion der WEB-Basis der gesamte Ablauf der
30 FE-Simulation verfolgt werden kann.

In der oberen Bildhälfte ist eine elastische Membran gegeben, die entlang dem Rand Γ eines ebenen Simulationsgebietes Ω fixiert ist und auf die im Inneren des Gebietes ein konstanter Druck $f = 1$ wirkt. Die Verschiebung u erfüllt bei geeigneter Normierung die Poisson-Gleichung mit homogenen Randbedingungen,

5
$$-\Delta u = 1 \quad \text{in } \Omega$$
$$u = 0 \quad \text{auf } \Gamma.$$

Die Verschiebung u bzw. Auslenkung der Membran ist in der unteren Bildhälfte der Fig. 5 ersichtlich. Wie oben beschrieben, gliedert sich das WEB-Verfahren in
10 die folgenden Schritte.

Eingabe 1 des Simulationsgebietes Ω : Der Rand Γ ist eine periodische Spline-Kurve vom Grad 6, die durch ihre Kontrollpunkte 20 (in Fig. 5 mit schwarzen Punkten gekennzeichnet) gespeichert wird.

15

Eingabe 2 der Randbedingungen: Die homogene Randbedingung ist wesentlich, so dass die Konstruktion einer Gewichtsfunktion erforderlich ist.

Eingabe 3 der Steuerparameter: Es wird der Grad $n = 2$ und, um die Figuren
20 übersichtlich zu gestalten, eine relativ große Gitterweite $h = 1/3$ verwendet.

Bestimmung 4 der Zelltypen: Wie in Fig. 6 illustriert, wird das Simulationsgebiet durch ein Gitter 21 überdeckt, das die Gitterzellen der Träger aller potenziell für die Basiskonstruktion relevanten B-Splines enthält. Die Typbestimmung ergibt in
25 dem Beispiel 69 äußere Gitterzellen 22 und 11 innere Gitterzellen 24, sowie 20 Gitterzellen 23 auf dem Rand.

Klassifikation 5 der B-Splines: Der Träger des B-Splines b_k ist hier das Quadrat Q_k mit den Ecken

30
$$(k_1, k_2)h, (k_1 + 3, k_2), (k_1 + 3, k_2 + 3)h, (k_1, k_2 + 3)h;$$

$Q_{(-4,0)}$ und $Q_{(2,1)}$ sind in Fig. 8 gezeigt. Die Gitterpunkte kh der relevanten B-Splines, für die Q_k das Innere des Simulationsgebietes schneidet, sind in Fig. 8 durch einen Punkt oder einen Kreis markiert. Alle Gitterpunkte ih für innere B-Splines (i aus der Indexliste I), für die mindestens eine Zelle des Trägers Q_i vollständig im Inneren von Ω liegt, sind durch einen Punkt markiert. Beispielsweise liegt für $i = (-4, 0)$ die Gitterzelle $(-2, 0)h + [0, h]^2$ ganz in Ω . Alle Gitterpunkte jh für äußere B-Splines (j aus der Indexliste J), für die keine Zelle des Trägers Q_j ganz in Ω liegt, sind durch einen Kreis markiert.

- 10 Berechnung 6 der Kopplungskoeffizienten: Zur Bestimmung der Kopplungskoeffizienten $e_{i,j}$ wird für jedes feste j der Indexliste J der nächstgelegene 3×3 -Bereich

$$I(j) = \{\ell_1, \ell_1 + 1, \ell_1 + 2\} \times \{\ell_2, \ell_2 + 1, \ell_2 + 2\}$$

- von Indizes in I gesucht. In Fig. 10 ist für den mit einem Kreis markierten äußeren Gitterpunkt $j = (-1, 2)$ der Bereich $I(j)$ durch Punkte gekennzeichnet. Die Figur 10 zeigt ebenfalls die zugehörigen Kopplungskoeffizienten in Matrixdarstellung. Sie werden mit bivariater Interpolation berechnet. Beispielsweise ist das Interpolationspolynom zu $i = (-1, -1)$

$$p_i(x) = -x_1(x_1 + 2)x_2(x_2 - 1)/2.$$

- 20 Sein Wert an der Stelle $x = j = (-1, 2)$ ist $p_i(j) = 1 = e_{i,j}$. Es fällt auf, dass viele der Kopplungskoeffizienten 0 sind. Dies ist ein typisches Phänomen. Die Kopplungskoeffizienten $e_{i,j}$ sind nur dann für alle i der Indexliste $I(j)$ ungleich 0, wenn die Indizes i in jeder Komponente von dem Index j verschieden sind.
- 25 Berechnungsvorschrift 7 für die Gewichtsfunktion: Die Gewichtsfunktion ist durch die Gleichung (3) gegeben mit $n = 2$ und $\delta = 0.2$. Der Parameter δ wird numerisch berechnet. Er muss klein genug sein, so dass Singularitäten der Abstandsfunktion vermieden werden. Zur Berechnung der Abstandsfunktion erstellt das Verfahren ein Programm, das mit Hilfe des Newton-Verfahrens arbeitet. Da die

Gewichtsfunktion nur in einem Randstreifen ungleich 1 ist, ist der Aufwand bei der späteren Auswertung gering.

Ausgabe: Fig. 13 zeigt den Träger eines WEB-Splines B_i und die zu seiner

- 5 Beschreibung benötigten Daten. Dies sind die Indexliste $J(i)$ der mit b_i gekoppelten äußeren B-Splines b_j , die Kopplungskoeffizienten $e_{i,j}$ und der Gewichtspunkt x_i . Diese Daten werden unter Einbeziehung der Gewichtsfunktion zur Erstellung einer Berechnungsvorschrift für die WEB-Splines benutzt.

- 10 Der weitere Ablauf der FE-Simulation folgt dem Stand der Technik.

Aufstellen 9 des FE-Systems: Die Einträge der Systemmatrix und der rechten Seite sind

$$G_{k,i} = \int_{\Omega} \text{grad } B_k \text{ grad } B_i, \quad F_k = \int_{\Omega} f B_k, \quad k, i \in I.$$

- 15 Das Gleichungssystem $GC = F$ für die Basiskoeffizienten C_i hat im betrachteten Beispiel die Dimension 31. Die Matrixeinträge $G_{k,i}$ werden mit Hilfe von numerischer Integration berechnet, ebenso die Integrale F_k .

Lösen 10 des FE-Systems: Das Galerkin-System wird iterativ mit dem Verfahren

- 20 der konjugierten Gradienten gelöst, wobei die SSOR-Vorkonditionierung zur Konvergenzbeschleunigung Verwendung findet. Nach 24 Iterationsschritten wird die Lösung im Rahmen der Rechnergenauigkeit gefunden (Toleranz $\leq 1e-14$).

Berechnung 11 und Ausgabe 12 der Approximation: Die mit dem

- 25 erfindungsgemäßen Verfahren berechnete Approximation ist $u = \sum_i C_i B_i$ und wird in in der unteren Bildhälfte von Fig. 5 grafisch dargestellt. Der relative Fehler in der L_2 -Norm beträgt 0.028.

Die Leistungsfähigkeit des erfindungsgemäßen Verfahrens in der speziellen

- 30 bevorzugten Ausführungsart (WEB-Verfahren) wird in einem zweiten

Ausführungsbeispiel anhand der Simulation einer inkompressiblen Strömung veranschaulicht. Die in Fig. 14 gezeigte Anordnung zweier kreisförmiger Hindernisse in einem parallel berandeten Kanal dient der Illustration des prinzipiellen Vorgehens. Bei komplizierteren Geometrien, wie sie in konkreten

- 5 Anwendungen typischerweise vorliegen, arbeitet das Verfahren vollkommen analog und effizient. In Fig. 14 sind die Strömungslinien 25 innerhalb des durch Γ_1 bis Γ_4 sowie durch Γ_5 und Γ_6 berandeten Gebiets dargestellt. Die Differenzialgleichung ist

$$\Delta u = 0 \quad \text{in } \Omega$$

mit den Randbedingungen

$$10 \quad \frac{\partial u}{\partial n} = v_0 \quad \text{auf } \Gamma_1, \quad \frac{\partial u}{\partial n} = -v_0 \quad \text{auf } \Gamma_2, \quad \frac{\partial u}{\partial n} = 0 \quad \text{auf } \Gamma_3, \dots, \Gamma_6.$$

Die Flussgeschwindigkeit $v = -\text{grad } u$ ist in der unteren Bildhälfte dargestellt.

- Die erfindungsgemäße Konstruktion der WEB-Basis läuft vollkommen analog zu dem ersten Ausführungsbeispiel ab. Der einzige Unterschied ist, dass aufgrund der
15 natürlichen Randbedingungen keine Gewichtsfunktion erforderlich ist.

- Fig. 15a bis Fig. 15c zeigen die Klassifikation der relevanten B-Splines für verschiedene Grade n (s. auch Fig. 8). In der Abbildung sind die inneren B-Splines b_i , die in die WEB-Basis ohne Erweiterung übernommen werden, durch schwarze
20 Dreiecke markiert. Für kleines h nimmt die Anzahl dieser B-Splines zu, d.h., es gilt $B_i = b_i$ für den überwiegenden Teil der WEB-Basis. Im betrachteten Beispiel trifft dies bei Grad $n = 3$ für 236 von 252 Indizes i zu.

- Fig. 16a zeigt in zwei Diagrammen den numerisch bestimmten relativen L_2 -Fehler
25 des Potenzials (linke Bildhälfte) in Abhängigkeit von der Gitterweite $h = 2^{-k}$ mit $k = 1, \dots, 5$ und die numerisch geschätzte Konvergenzordnung m (rechte Bildhälfte). Dabei werden für die verschiedenen Grade der WEB-Splines als Kennzeichner $*$ ($n = 1$), \circ ($n = 2$), \triangle ($n = 3$), \square ($n = 4$) und \star ($n = 5$) verwendet. Erwartungsgemäß ergibt sich $m \approx n + 1$, d.h. eine ungefähre Fehlerreduktion um

den Faktor 2^{n+1} bei Halbierung der Gitterweite. Analog ergibt sich für den in der Fig. 16b abgebildeten relativen Approximationsfehler der Flussgeschwindigkeit (H^1 -Norm der Lösung, linke Bildhälfte) eine Konvergenzordnung von $m \approx n$ (rechte Bildhälfte) und die damit verbundene Fehlerreduktion um etwa den Faktor 5 2^n bei Halbierung der Gitterweite.

Fig. 16c (rechte Bildhälfte) zeigt die Rechenzeit in Sekunden für die Konstruktion der WEB-Basis in Abhängigkeit von der Zahl der resultierenden Basisfunktionen gemessen auf einem Pentium II - Prozessor mit 400 MHz. Beispielsweise werden
10 für die Konstruktion einer WEB-Basis vom Grad 3 zur Gitterweite $h = 0.125$ mit 2726 WEB-Splines 1.32 Sekunden benötigt. Es fällt auf, dass der Aufwand zum Erstellen der WEB-Basis weitgehend unabhängig vom Grad n der Basis ist. In der linken Bildhälfte der Fig. 16c ist die Anzahl der CG-Iterationen bezogen auf die Anzahl der Basisfunktionen gezeigt. So werden für das zugehörige System mit 2726
15 Unbekannten 65 PCG-Iterationen benötigt. Die Gesamtrechenzeit inklusive Aufstellen und Lösen des Galerkin-System beträgt etwa 2.48 Sekunden.

In Fig. 17 wird die WEB-Methode mit einem Standardlösungsverfahren, das das Simulationsgebiet vernetzt bzw. trianguliert (obere Bildhälfte) und Hutfunktionen
20 verwendet, verglichen. Die Grafik zeigt in der unteren Bildhälfte den L_2 -Fehler bezogen auf die Anzahl der Parameter. Die Ergebnisse des Standardlösers sind mit fetten Karo-Kennzeichnern markiert und werden den mit Hilfe von WEB-Basen der Grade 1 bis 5 erzielten Ergebnissen gegenübergestellt. Beispielsweise wird eine Genauigkeit von 10^{-2} mit der WEB-Methode durch Verwendung von 213
25 Basisfunktionen mit Grad 2 erreicht bei einer Gesamtrechenzeit von 0.6 Sekunden. Zum Erzielen der gleichen Genauigkeit benötigt das Standardverfahren mit linearen Hutfunktionen 6657 Basisfunktionen.

Bei der Beurteilung des Standardlösungsverfahrens sind noch zwei weitere Aspekte zu beachten. Zum einen verdeutlicht Fig. 17, dass selbst eine moderate Genauigkeit von 10^{-3} mittels Hutfunktionen nur erreichbar ist, wenn weit mehr als eine Million Koeffizienten verwendet werden. Dies zeigt, dass genaue Resultate bei der Verwendung von Hutfunktionen in der Regel eine enorme Rechen- und Speicherkapazität erfordern oder aber nach dem Stand der Technik gar nicht erreichbar sind. Zum anderen steigt der zur Vernetzung erforderliche Aufwand mit der Komplexität des Simulationsgebietes. Im Gegensatz zu realistischen Anwendungen ist das hier untersuchte Gebiet aufgrund seiner sehr einfachen Struktur noch vergleichsweise einfach zu triangulieren.

Das zweidimensionale Beispiel zeigt den Performancegewinn durch das WEB-Verfahren. Eine noch größere Leistungssteigerung ist bei dreidimensionalen Problemen möglich. Zum einen ist dort der Aufwand für die Vernetzung, die bei dem WEB-Verfahren entfällt, wesentlich größer. Zum anderen macht sich hier auch die Reduktion der Anzahl der benötigten Basisfunktionen noch wesentlich deutlicher bemerkbar als im zweidimensionalen Fall.

Die Fig. 18 zeigt eine erfindungsgemäße Vorrichtung, insbesondere eine Computereinrichtung 30, mit Eingabemitteln 31, 32, 33, Ausgabemitteln 34 sowie eine den Verfahrensablauf steuernde Steuereinrichtung 35. Zur Abarbeitung des erfindungsgemäßen Verfahrens und insbesondere zum Zwecke der Parallelisierung des zugehörigen Rechenablaufs bedient sich die zentrale Steuereinrichtung 35 vorzugsweise mehrerer arithmetischer logischer Einheiten (Arithmetic Logical Unit, ALU) oder sogar mehrerer zentraler Verarbeitungseinheiten (Central Processing Unit, CPU) 36. Diese erlauben insbesondere eine parallele Abarbeitung der Verfahrensschritte Klassifizieren 5 der B-Splines, insbesondere auch Schneiden des regulären Gitters mit dem Simulationsgebiet Ω , Ermitteln 6 der Kopplungskoeffizienten $e_{i,j}$ und/oder Auswerten der Gewichtsfunktion $w(x)$ an Punkten x des Simulationsgebietes Ω .

Die Recheneinheiten 36 greifen dabei auf den gemeinsamen Datenbestand von Speichermitteln 37 zu. Die Eingabe der Daten kann beispielsweise mittels einer Tastatur 31, einem maschinenlesbaren Datenträger 38 über eine zugehörige Lesestation 32 und/oder über ein drahtgebundenes oder drahtloses Datennetz mit
5 einer Empfängerstation 33 erfolgen. Über die Lesestation 32 bzw. einen zugehörigen Datenträger 38 kann auch das den Verfahrensablauf steuernde Steuerprogramm eingegeben werden und beispielsweise auf dem Speichermittel 37 dauerhaft abgelegt werden. Entsprechend kann das Ausgabemittel 34 einen Drucker, einen Monitor, eine Schreibstation für einen maschinenlesbaren Datenträger und/oder
10 eine Sendestation eines drahtgebundenen oder drahtlosen Datennetzes aufweisen.

Patentansprüche

1. Verfahren zur Erhöhung der Leistungsfähigkeit einer Computereinrichtung zur Finiten-Elemente-Simulation durch automatische Generierung geeigneter Basisfunktionen mit Hilfe von B-Splines mit den folgenden Schritten:
 - 5 • Definieren (1) eines Simulationsgebietes (Ω) und Abspeichern von Gebietsdaten des Simulationsgebietes (Ω),
 - Eingabe (2) und Abspeichern von Randbedingungen,
 - Festlegen (3) einer vorgebbaren Gitterweite h und eines vorgebbaren Grades n der B-Splines,
 - 10 • Ermitteln (4) eines sich über das Simulationsgebiet (Ω) erstreckenden Gitters und eines Typs der Gitterzellen,
 - Klassifizieren (5) der B-Splines, deren Träger das Simulationsgebiet (Ω) schneiden, durch Ermitteln von inneren und äußeren B-Splines, wobei für äußere B-Splines der Schnitt des Trägers mit dem Simulationsgebiet
 - 15 (Ω) kleiner als eine vorgebbare Schranke s ist,
 - Ermitteln (6) von Kopplungskoeffizienten $e_{i,j}$ zur Bildung von Linearkombinationen von inneren und äußeren B-Splines,
 - Speicherung und Ausgabe der die Basisfunktionen bestimmenden Parameter.
- 20 2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass vor dem Speichern und Ausgeben der Parameter der Schritt durchgeführt wird:
Festlegen (7) einer vorgebbaren Gewichtsfunktion w sowie Bestimmen (8) von Gewichtspunkten und Skalierungsfaktoren.
3. Verfahren nach Anspruch 2, dadurch gekennzeichnet, dass die
- 25 Gewichtsfunktion w durch einen glatten Übergang von einem konstanten Plateau im Inneren des Simulationsgebietes (Ω) zum Wert Null auf dem Rand (Γ) festgelegt wird.

4. Verfahren nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, dass B-Splines mit mindestens einer ganz im Simulationsgebiet (Ω) enthaltenen Gitterzelle des Trägers als innere B-Splines klassifiziert werden.
5. Verfahren nach einem der Ansprüche 1 bis 4, dadurch gekennzeichnet, dass der Gewichtspunkt als Mittelpunkt einer ganz im Simulationsgebiet (Ω) enthaltenen Gitterzelle des Trägers des entsprechenden inneren B-Splines gewählt wird.
6. Verfahren nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, dass das Simulationsgebiet (Ω) durch Einspeichern von aus einer
10 computergestützten Konstruktion (CAD/CAM) ableitbaren Daten definiert wird.
7. Verfahren nach einem der Ansprüche 1 bis 6, dadurch gekennzeichnet, dass die Gitterweite h unter Verwendung von abgespeicherten empirisch und/oder analytisch gewonnenen Werten mittels einer zugehörigen ersten
15 Bewertungsfunktion automatisch festgelegt wird.
8. Verfahren nach einem der Ansprüche 1 bis 7, dadurch gekennzeichnet, dass der Grad n unter Verwendung von abgespeicherten empirisch und/oder analytisch gewonnenen Werten mittels einer zugehörigen zweiten Bewertungsfunktion automatisch festgelegt wird.
- 20 9. Verfahren nach einem der Ansprüche 1 bis 8, gekennzeichnet durch die Schritte:
 - Aufstellen (9) eines bei der FE-Simulation zu lösenden Gleichungssystems,
 - Lösen (10) des Gleichungssystems,
 - 25 • Berechnen (11) einer Näherungslösung, und
 - Ausgeben (12) der Näherungslösung.

10. Verfahren nach Anspruch 9, dadurch gekennzeichnet, dass beim Lösen (10) des Gleichungssystems ein Mehrgitter-Verfahren eingesetzt wird.
11. Vorrichtung zur Durchführung eines Verfahrens nach einem der Ansprüche 1 bis 10, insbesondere Computereinrichtung, mit Eingabemitteln (31, 32, 33) und Ausgabemitteln (34), Speichermitteln (37) und einer Recheneinheit (35, 36), dadurch gekennzeichnet, dass die reguläre Gitterstruktur zur Optimierung des Rechenablaufs, insbesondere durch Parallelisierung, ausgenutzt wird.
12. Maschinenlesbarer Datenträger (38), insbesondere Magnetband, Magnetscheibe, Compact Disc (CD) oder Digital Versatile Disc (DVD), dadurch gekennzeichnet, dass auf dem Datenträger (38) ein Steuerprogramm für eine Computereinrichtung (30) gespeichert ist, gemäß dem durch die Computereinrichtung (30) ein Verfahren nach einem der Ansprüche 1 bis 10 abgearbeitet werden kann.

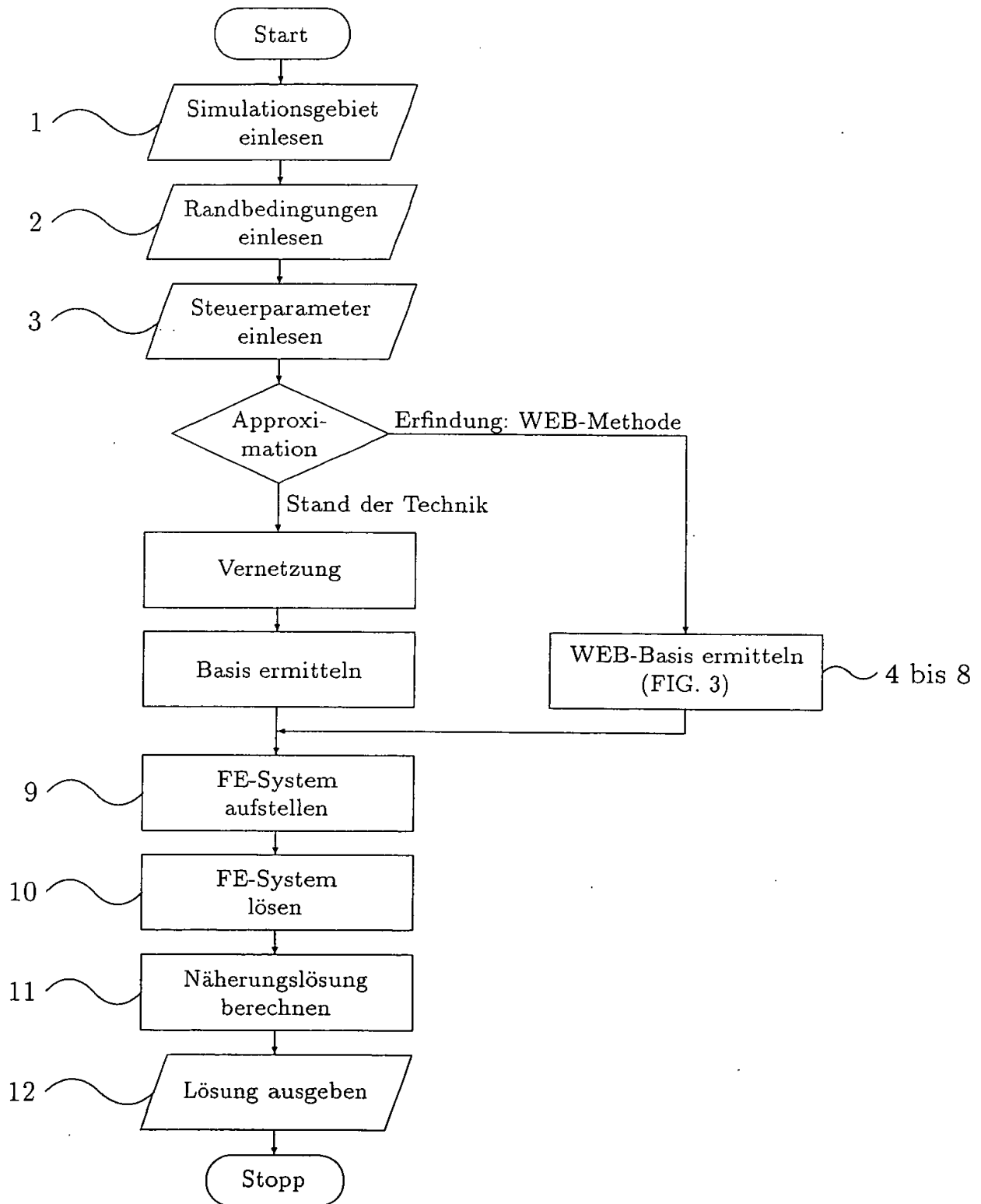
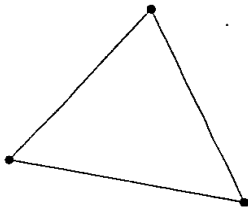
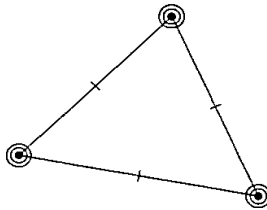


Fig. 1

lineares Element



Agyris-Dreieck



Clough-Tocher Element

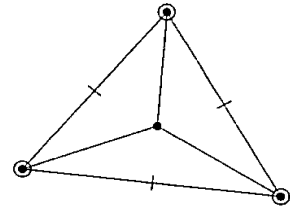
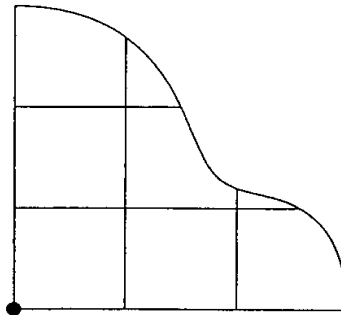


Fig. 2a
Stand der Technik

WEB-Spline



Typ	Dimension	Grad	Glattheit	Parameter
lineares Element	d	1	0	$d + 1$
Agyris-Dreieck	2	5	1	21
Clough-Tocher Element	2	3	1	12
WEB-Spline	d	n	$n - 1$	1

Fig. 2b

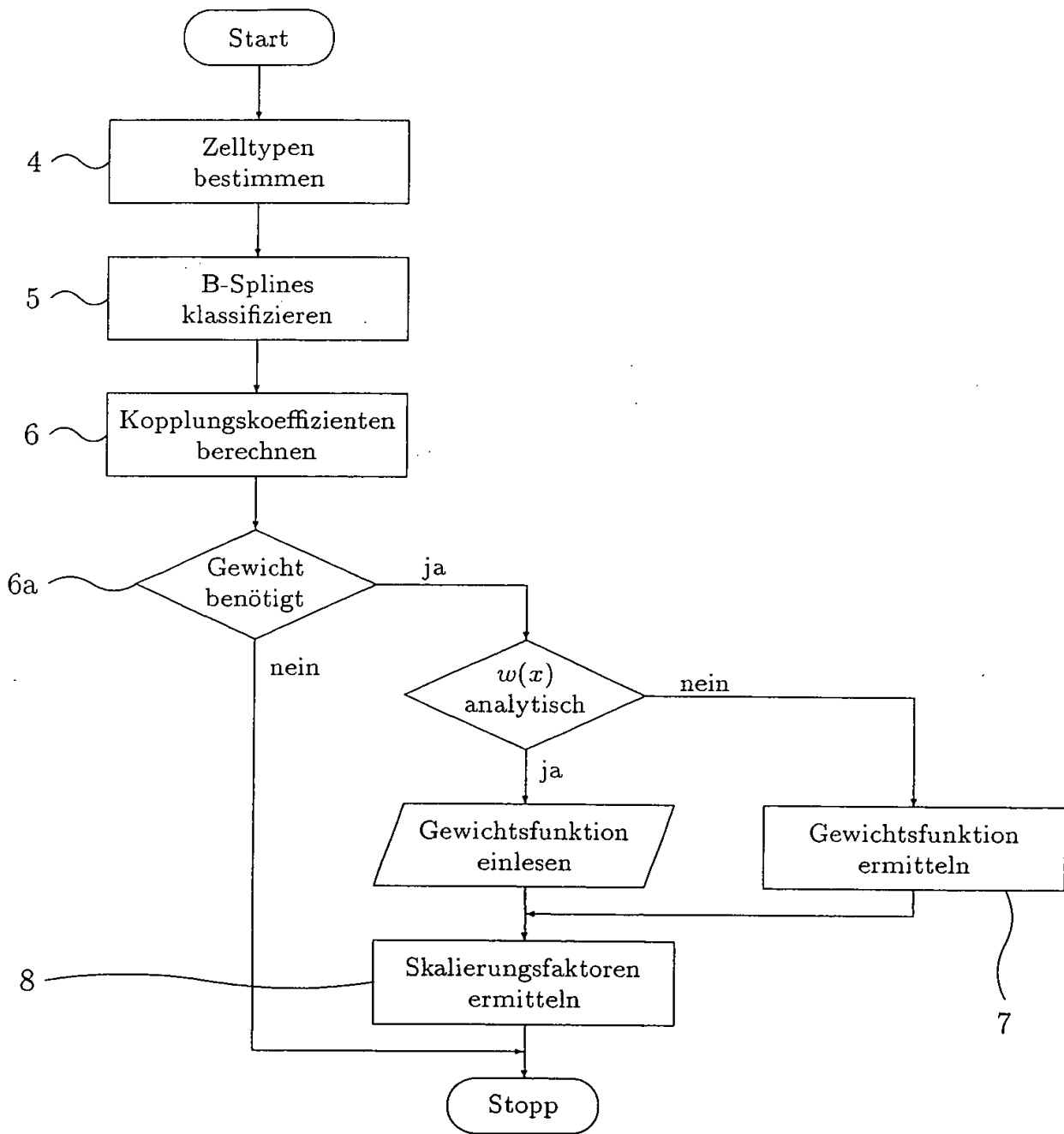


Fig. 3

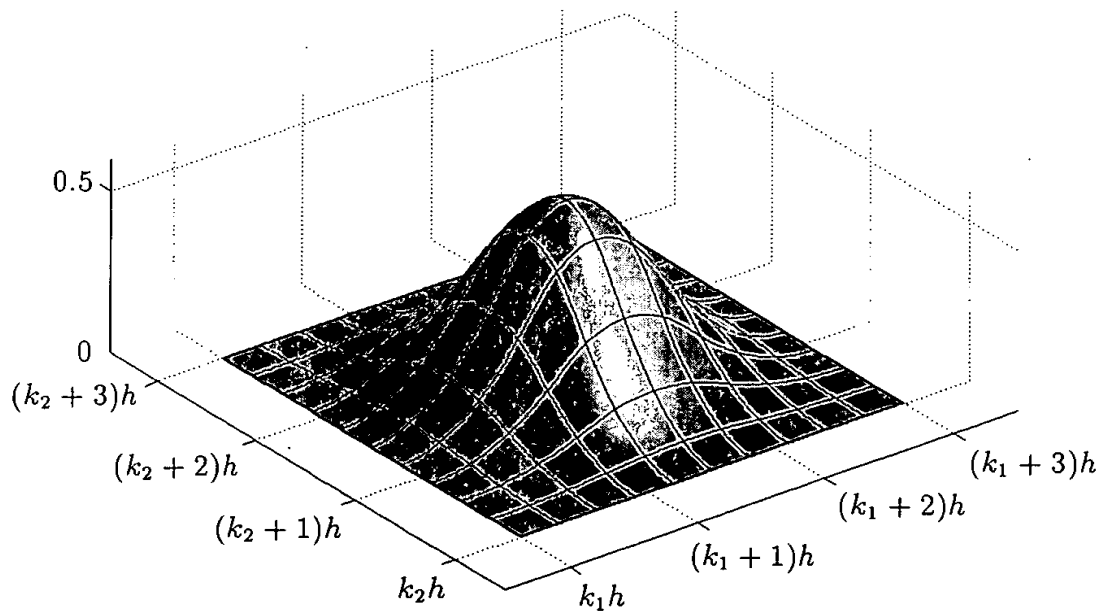
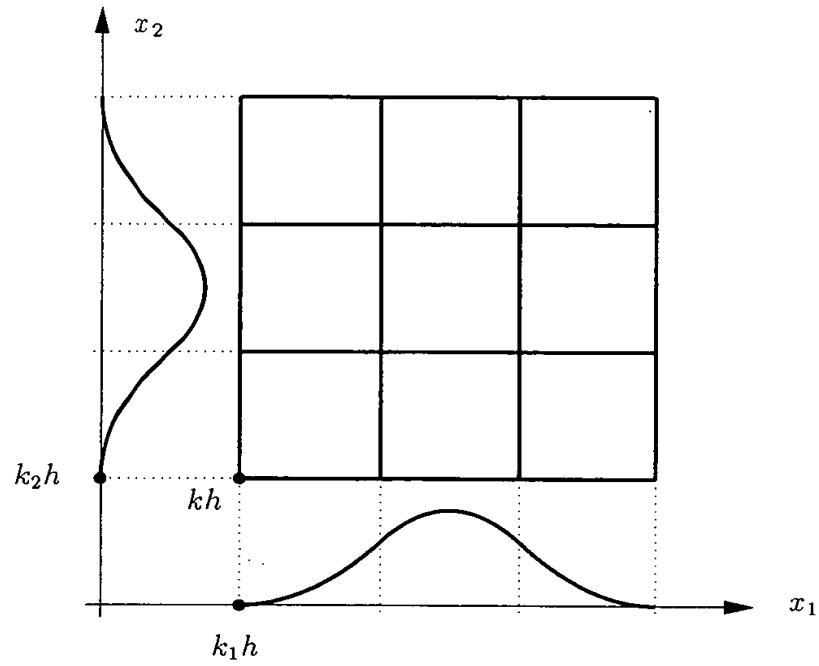


Fig. 4

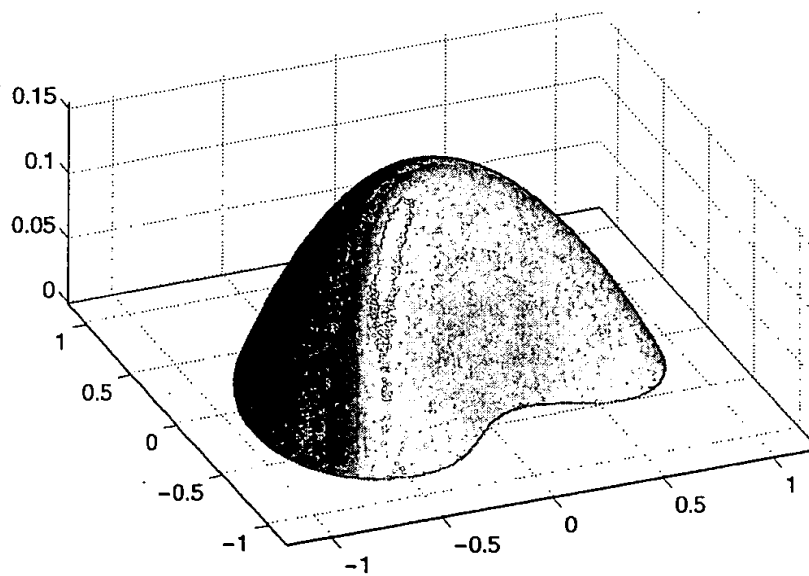
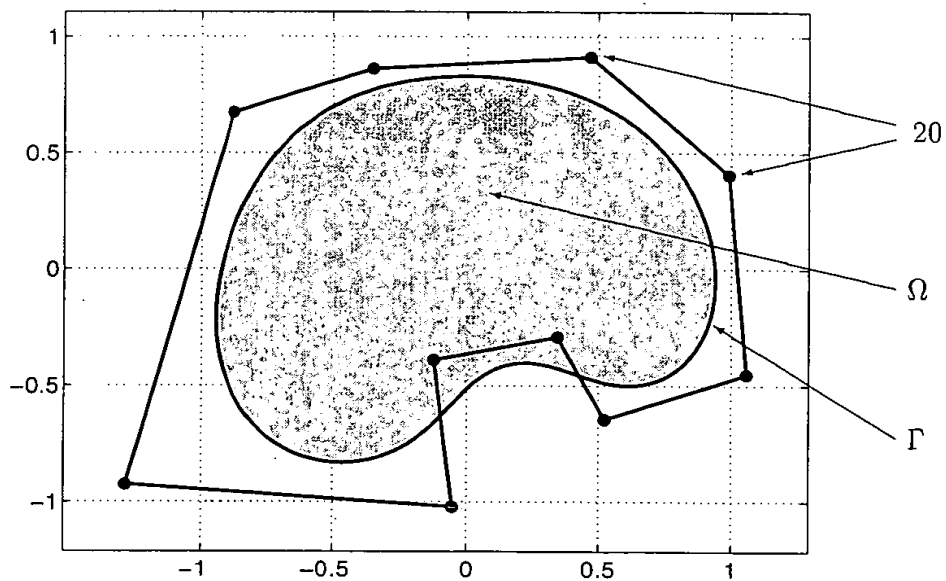


Fig. 5

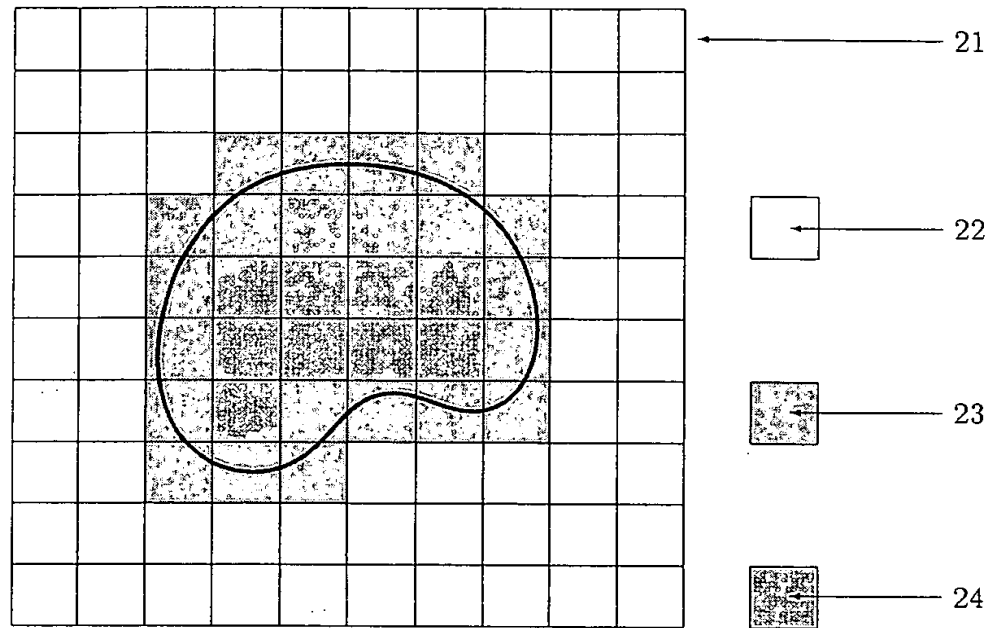


Fig. 6

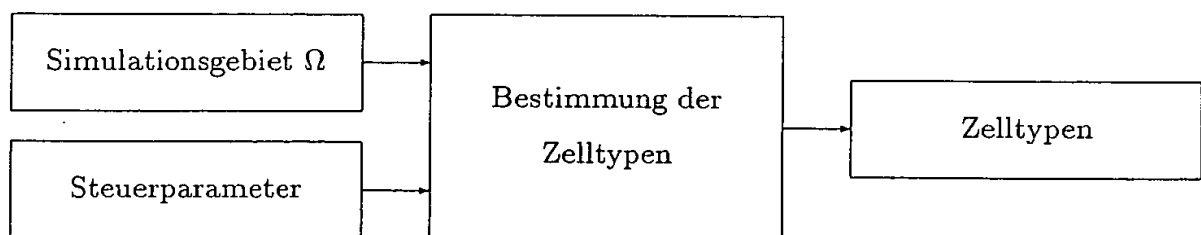


Fig. 7

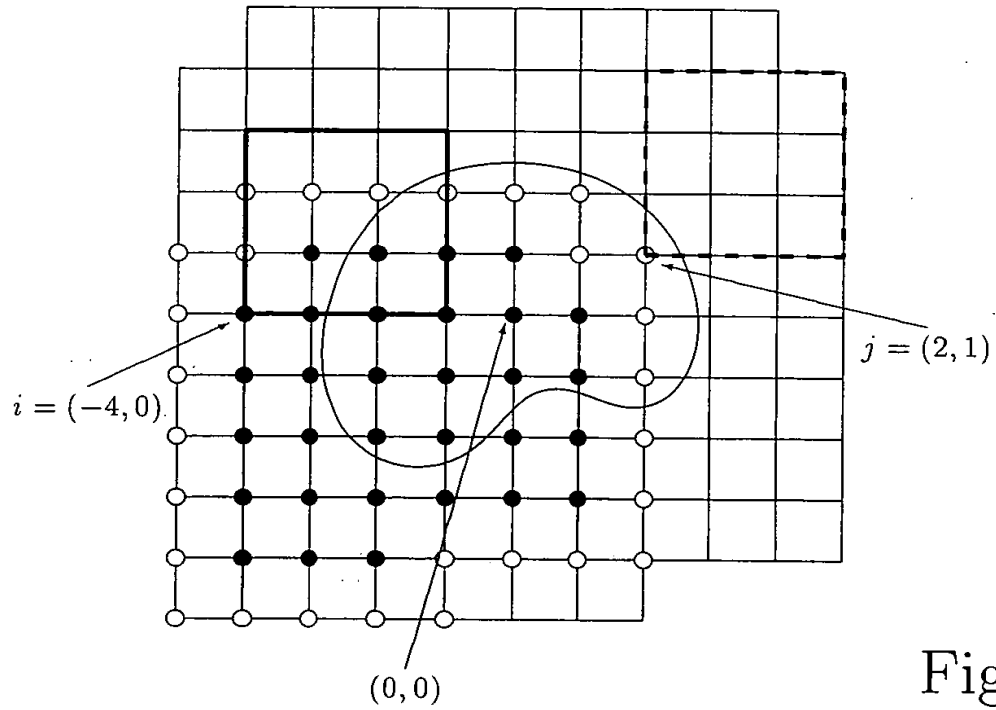


Fig. 8

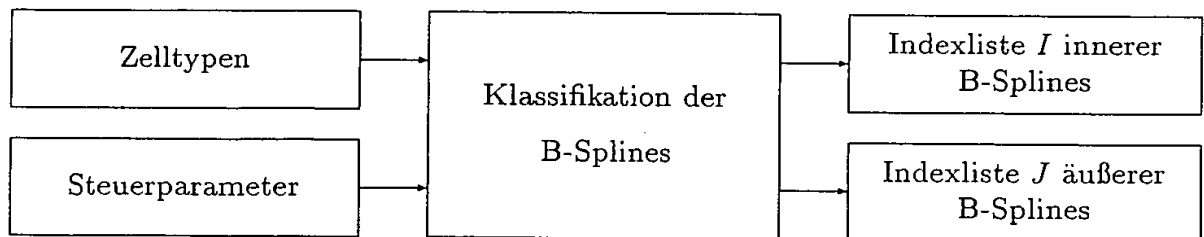
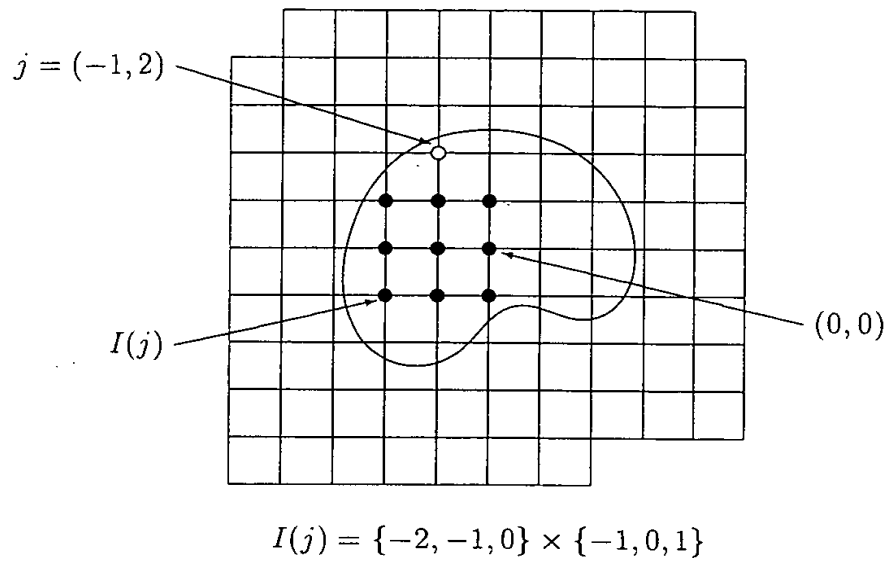


Fig. 9



$$e_{i,j} : \begin{matrix} 0 & \dots & -3 & \dots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \dots & -3 & \dots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \dots & 1 & \dots & 0 \end{matrix}$$

Fig. 10

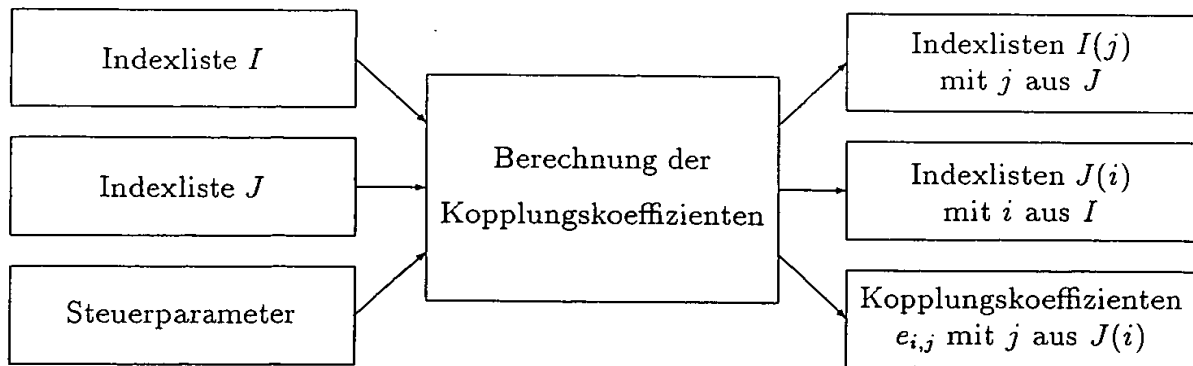


Fig. 11

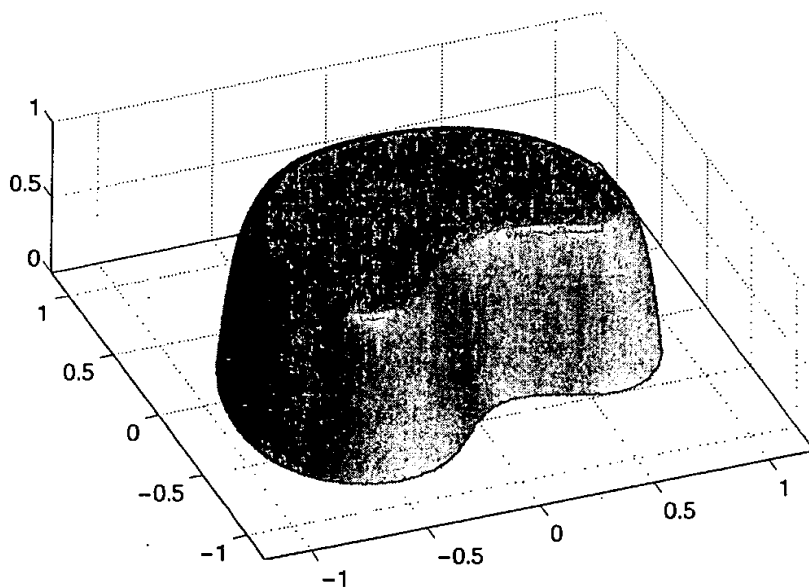
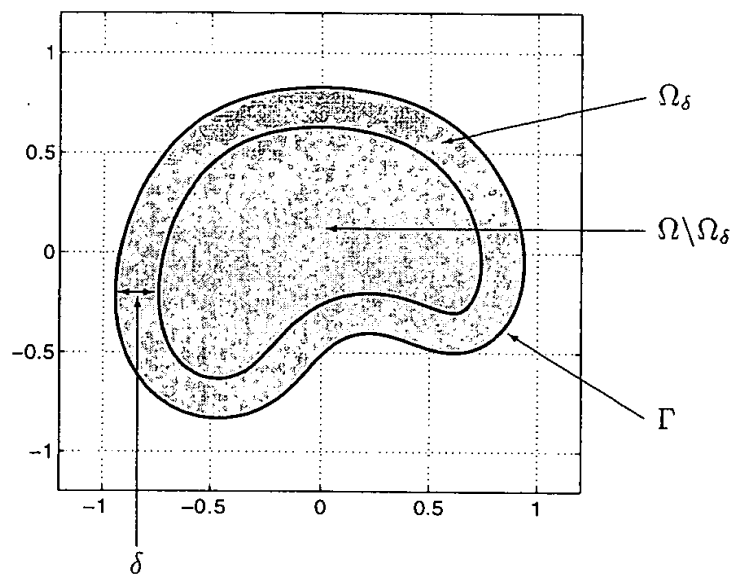
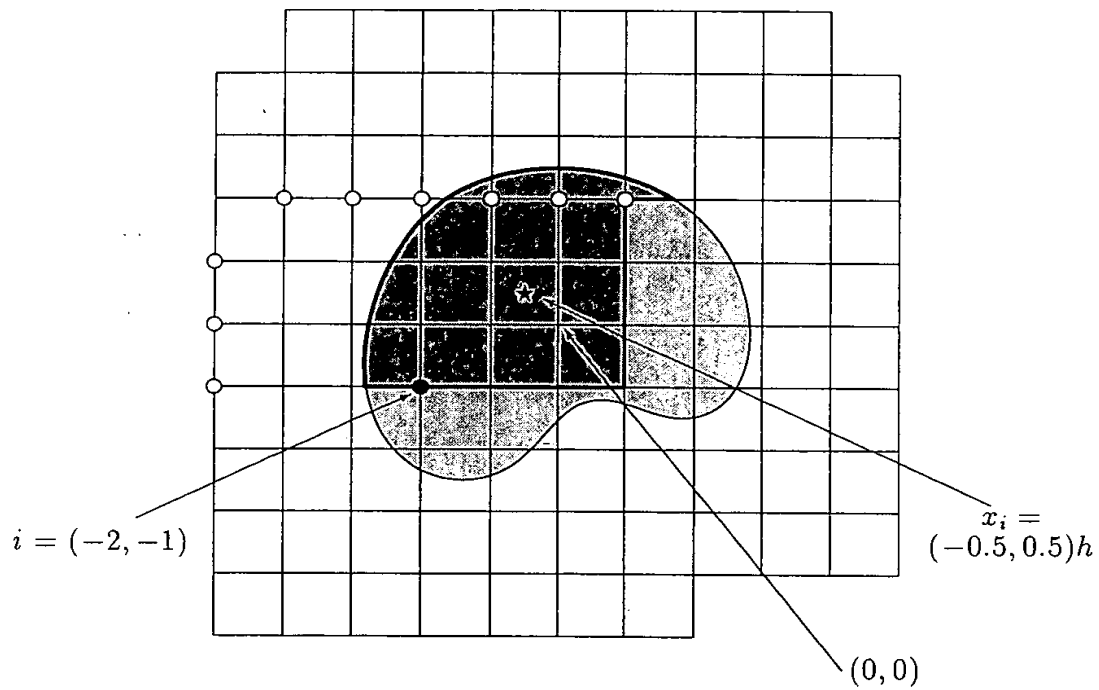


Fig. 12



$$J(i) = \{(-5, -1), \dots, (-5, 1), (-4, 2), \dots, (1, 2)\}$$

$$e_{i,j} : \begin{array}{cccccccccccc} \cdot & \dots & -3 & \dots & 0 & \dots & 1 & \dots & 0 & \dots & 0 & \dots & 1 \\ \vdots & & & & & & & & & & & & \\ -3 & & & & & & & & & & & & \\ \vdots & & & & & & & & & & & & \\ 0 & & & & & & & & & & & & \\ \vdots & & & & & & & & & & & & \\ 1 & & & & & & & & & & & & \end{array}$$

Fig. 13

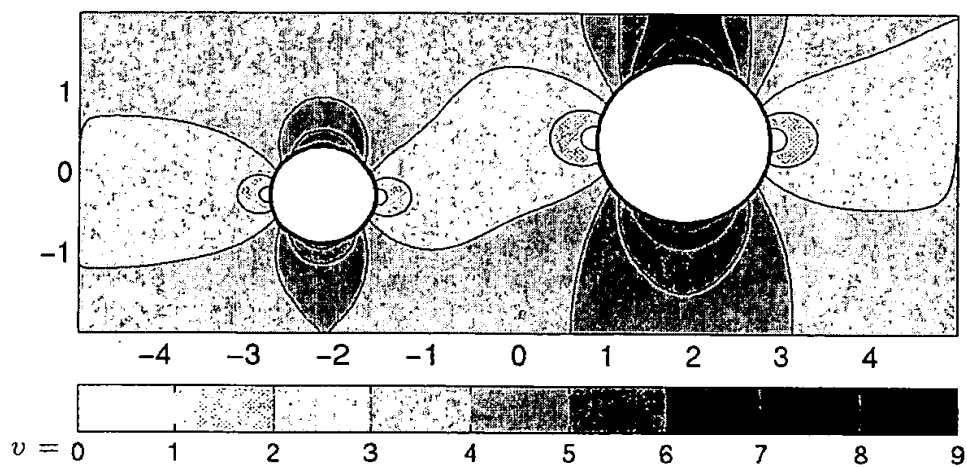
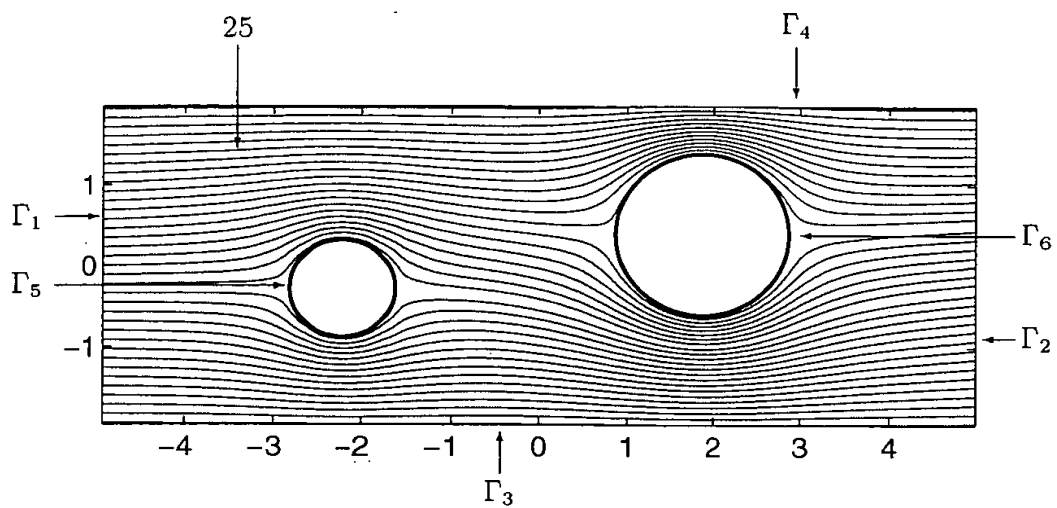


Fig. 14

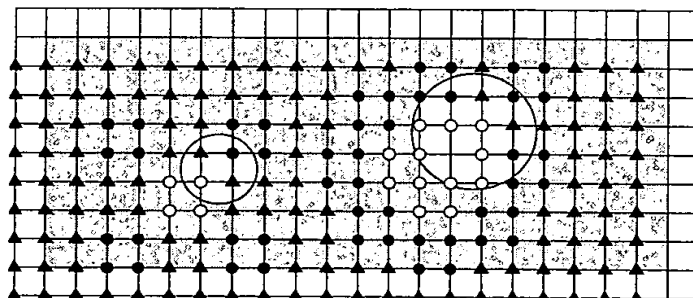


Fig. 15a

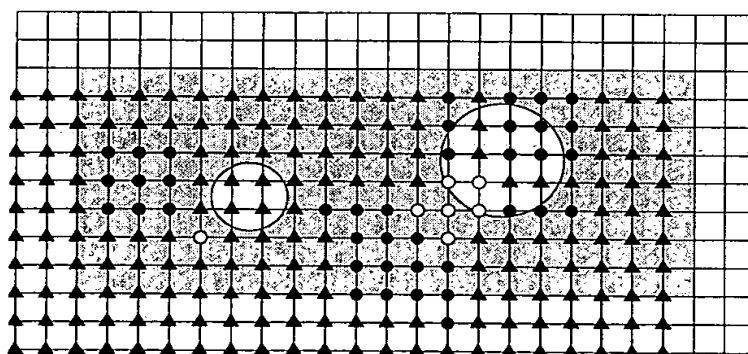


Fig. 15b

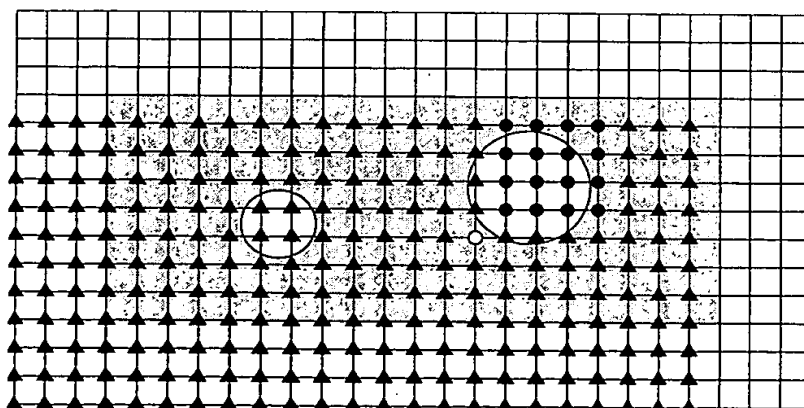


Fig. 15c

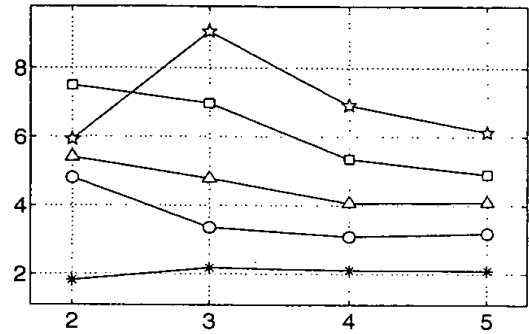
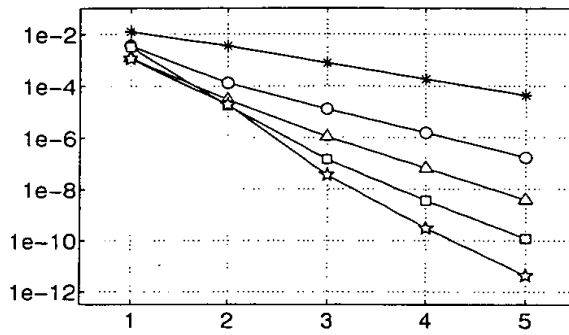


Fig. 16a

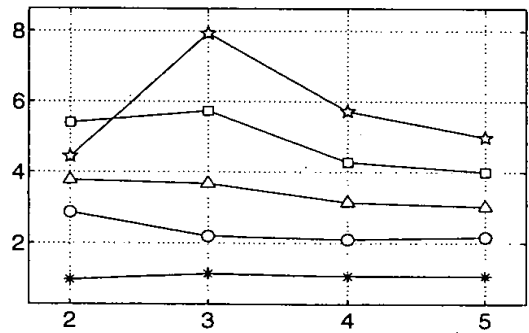
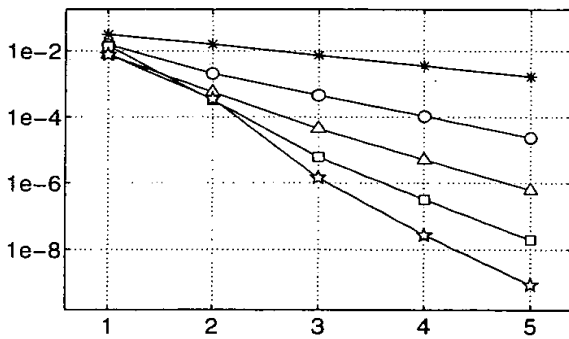


Fig. 16b

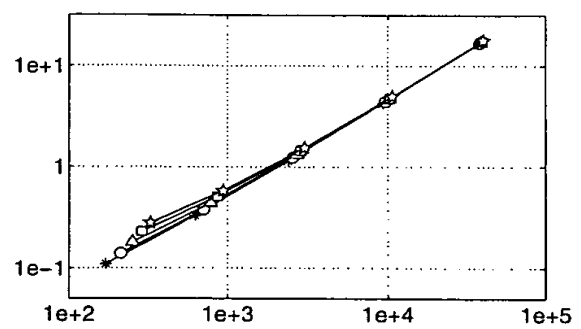
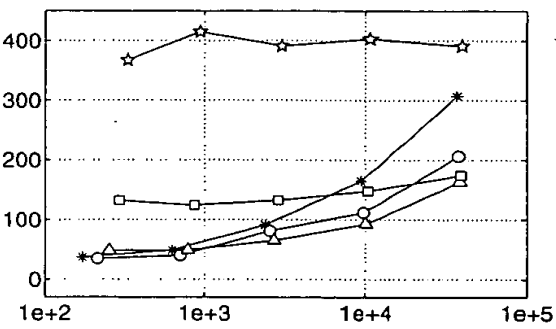


Fig. 16c

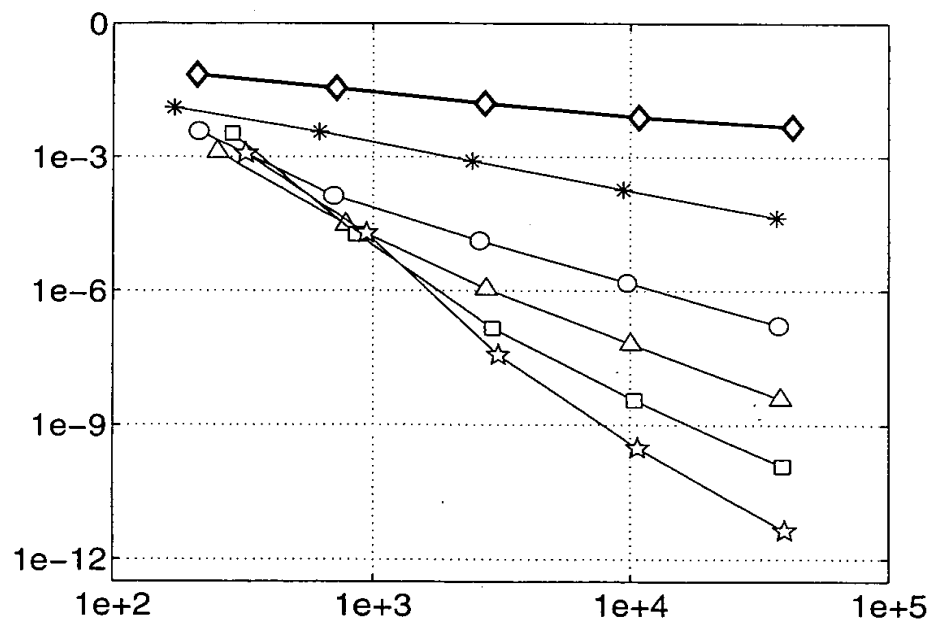
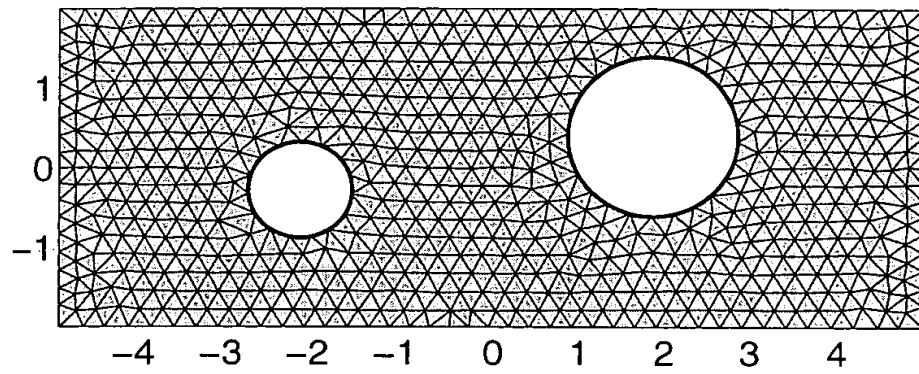


Fig. 17

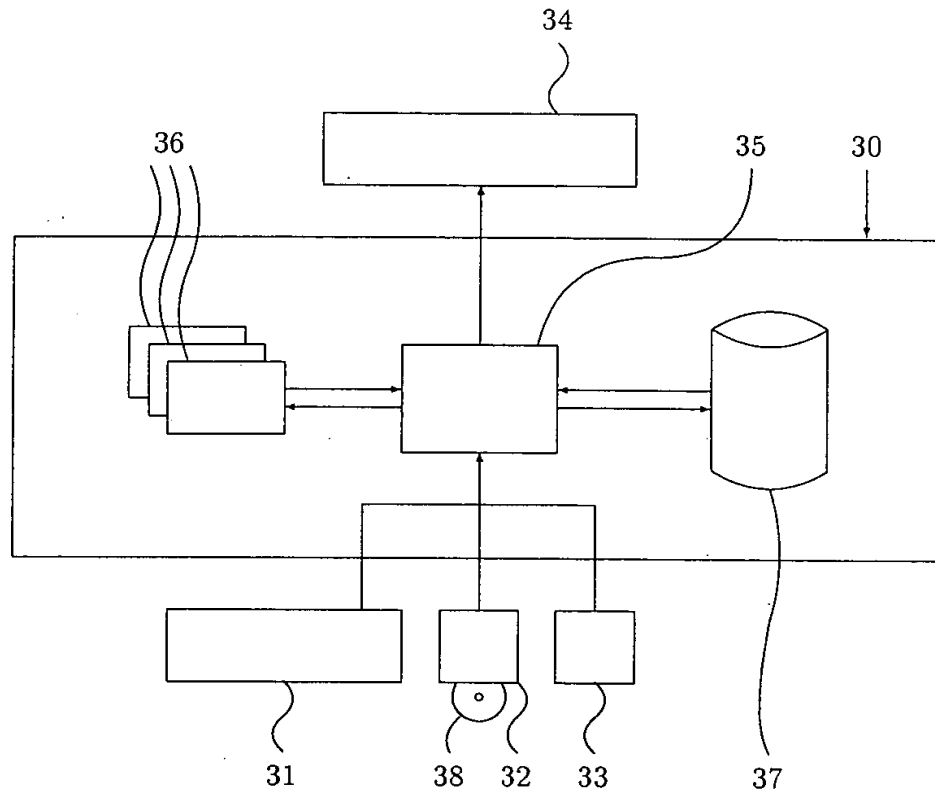


Fig. 18

Zusammenfassung

Die Erfindung betrifft ein Verfahren zur Erhöhung der Leistungsfähigkeit einer Computereinrichtung bei Finite-Elemente-Simulationen durch effiziente automatische Konstruktion geeigneter Basisfunktionen für die Berechnung von Näherungslösungen, und eine solche Computereinrichtung. Bei dem

5 erfindungsgemäßen Verfahren wird ein das Simulationsgebiet Ω überdeckendes Gitter erstellt. Darauf definierte B-Splines, deren Träger das Simulationsgebiet schneiden, werden in innere und äußere B-Splines klassifiziert (5). Anschließend werden Kopplungskoeffizienten zur Bildung von Linearkombinationen von inneren und äußeren B-Splines ermittelt (6) und die resultierenden Basisfunktionen

10 bestimmenden Parameter gespeichert und ausgegeben.

Fig. 3

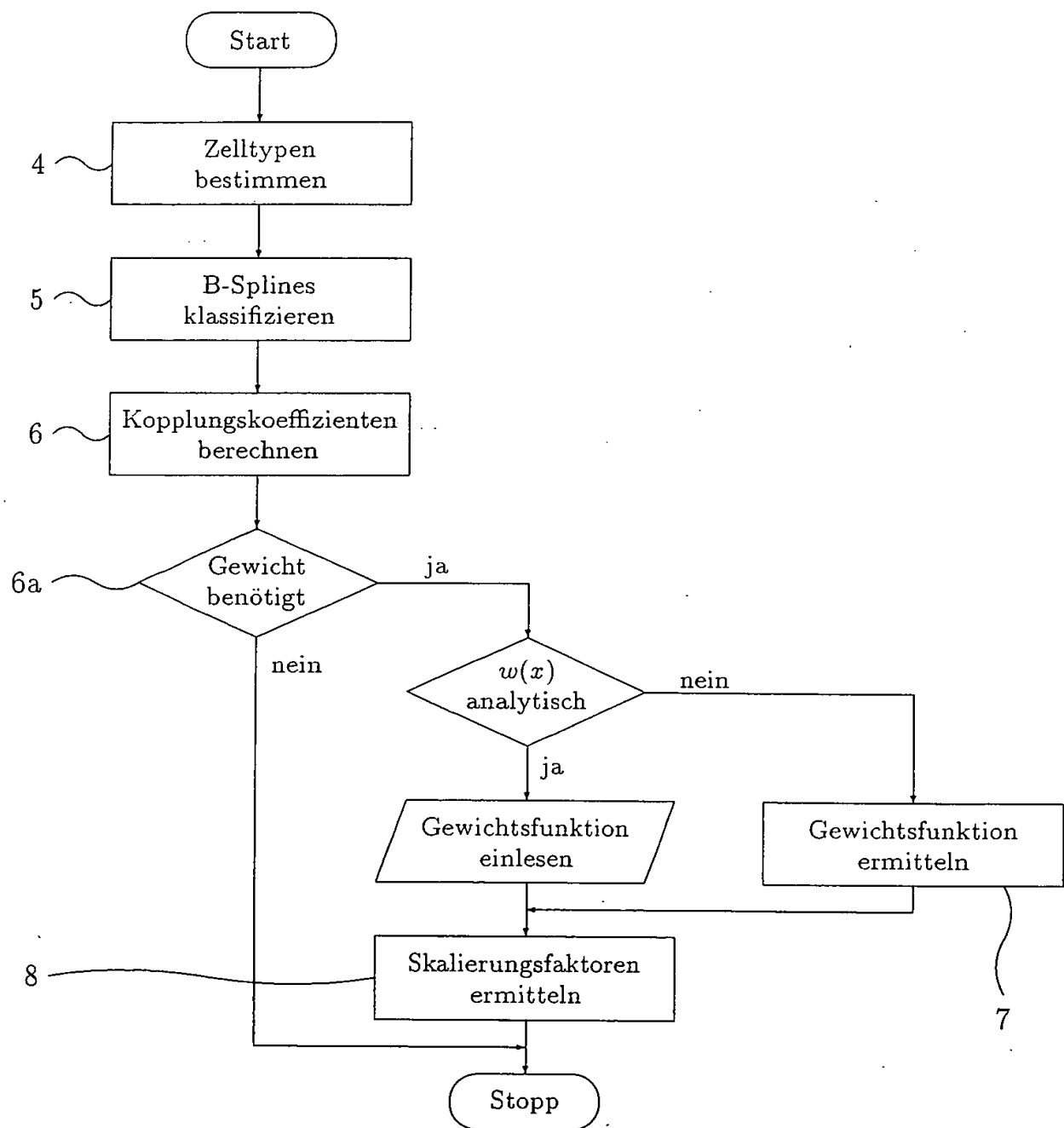


Fig. 3



Stuttgart, May 11, 2000

University of Stuttgart
Keplerstrasse 7, D-70174 Stuttgart

Process for Increasing the Efficiency
of a Computer System in Finite Element Simulations
and such a Computer System

The invention relates to a process for increasing the efficiency of a computer system in finite element simulations by efficient automatic construction of suitable basis functions for computation of approximate solutions, and to a computer system for performing that process.

5

A plurality of technical and physical phenomena can be described by partial differential equations. They include among others problems from fluid mechanics (for example, flow around an airfoil), electromagnetic field theory (for example, electrical field behavior in a transistor) or elasticity theory (for example, deformation of a car body). Accurate knowledge and description of such processes are a central element in the construction and optimization of technical objects. To save time-consuming and cost-intensive experiments, there is great interest in computer-aided simulations. Finite element processes (FE processes) have become established and have been the topic of intensive research for a long time. This applies especially also to automatic mesh generation processes as a foundation for construction of suitable basis functions.

10

15

Figure 1 illustrates in the left half the prior art in the process of FE simulation for a linear boundary value problem as a typical model example. Proceeding from the

data describing the geometry of the technical object to be simulated, first a system of basis functions is constructed which on the one hand enables fulfilment of boundary conditions and on the other is suited for approximation of the unknown solution. Then, using these basis functions a system of linear equations is set up
5 by numerical integration methods. Finally, the coefficients of the unknown approximation are determined as the solution of this system of linear equations.

FE methods or their use are the subject matter of a series of patents. For example, see US 4, 819, 161 where FE approximations of a large class of differential equation
10 are automated, and US 5, 731, 817 where a process for generation of hexahedral meshes forms the foundation for a FE simulation process.

In most FE processes of practical relevance basis functions are used which are defined on a decomposition produced by generating a mesh of the simulation
15 region. Figure 2a shows a selection of conventional elements; their dimension, degree, smoothness and parameters are listed in Figure 2b. A survey of meshing methods of planar regions can be found for example in K. Ho-Le, Finite Element Mesh Generation Methods: A review and classification. Comp. Aided Design 20 (1988), 27 – 38. Generating a mesh for complicated three-dimensional regions is
20 extremely difficult using the current state of knowledge, as shown by S. Owen, A survey of unstructured Mesh Generation Technology, Proceedings, 7th International Meshing Round Table, Sandia National Lab (1998), 239 – 257. The processes require extensive amounts of computing time and are to some extent not yet completely automated. But recently, there has been a series of very innovative
25 new approaches. For example, A. Fuchs, Optimierte Delaunay-Triangulierungen zur Vernetzung getrimmter NURBS-Körper, University of Stuttgart, 1999, simulates a force distribution in order to achieve an optimum distribution of triangulation points. In US 5, 729, 670 two- and three-dimensional meshes are produced by solving flow problems; this is an interesting reversal of the
30 conventional FE mechanism. In addition, many algorithms have been developed to

improve individual component aspects of mesh generation processes. For example DE 196 21 434 A1 and US 5, 774, 696 describe a process for elimination of intersections with prescribed edges or boundary surfaces in Delaunay triangulations.

5

Meshless FE-methods to date have not acquired any importance for applications. Both in the Lagrange multiplier method, see for example J.H. Bramble, The Lagrange Multiplier Method for Dirichlet's Problem, Math. Comp.

37 (1981), 1 – 11, and also in the penalty method, see for example P. Bochev and
10 M. Gunzburger, Finite Element Methods of Least Squares Type, SIAM review
40 (1998), 789 – 837, the treatment of boundary conditions represents a major
problem in the use of simple, stable basis functions.

In many technical simulations, automatic mesh generation is very complex and
15 requires by far the largest part of the computer time. Furthermore, the
approximation power of the conventionally used linear and multilinear basis
functions is low. To achieve accurate results, a large number of basis functions
must be used, and thus, a correspondingly large system of equations must be
solved. Higher order trial functions on triangulations generally likewise have an
20 unfavorable ratio between the attainable accuracy and the number of basis
functions used. Finally, smooth basis functions cannot be easily defined on
unstructured meshes. Very special constructions are necessary already for
continuously differentiable elements (see Figure 2a).

25 The object of the invention is to increase the efficiency of known FE methods and
computer systems which carry out FE methods by efficient construction of basis
functions with favorable properties. In particular the meshing of the simulation
region will be completely eliminated, optional boundary conditions are fulfilled,
accurate solutions are obtained with relatively few coefficients, and the resulting
30 system of equations will be solvable efficiently. In this way the disadvantages of

the prior art will be overcome, and thus, the accuracy and speed of the simulation of physical properties in the engineering and optimization of technical objects will be improved.

- 5 Some central terms and notations which are used in the following description of the process as claimed in the invention will be explained first.

The simulation region Ω is a bounded set of dimension $d = 2$ or $d = 3$ on which the physical quantities to be studied will be approximated by means of FE methods.

- 10 The boundary of the simulation region is denoted by Γ . A grid with grid width h is defined as a decomposition of a subset of the plane or the space in grid cells Z_k . Depending on the dimension d each grid cell is a square or a cube with edge length h . More precisely, $Z_k = kh + [0, h]^d$, where k belongs to a set of integer d -vectors. The uniform tensor product B-splines in d variables of degree n with grid width h
15 are denoted by b_k , see for example C. de Boor, A Practical Guide to Splines, Springer, 1978. They are functions which can be continuously differentiated $(n - 1)$ times and which on the grid cells agree with polynomials of degree n , as shown in Figures 4a and 4b. Figure 4a shows a support Q_k of the B-spline of degree $n = 2$, dimension $d = 2$ and smoothness $m = 1$. In Figure 4b, the resulting tensor product
20 B-spline b_k is shown. The support Q_k , i.e. the union of all grid cells, on which the B-spline b_k is not identically zero, consists of $(n + 1)^d$ grid cells; more precisely, $Q_k = kh + [0, (n + 1)h]^d$. In all figures, the B-spline b_k is marked at the point kh , i.e. for example in the case $d = 2$ at the lower left corner of the support. For FE simulations only those B-splines are important which have support intersecting the
25 simulation region Ω ; they are called relevant B-splines. The relevant B-splines are again divided into two groups; those B-splines for which the part of the support inside the simulation region is larger than a prescribed bound s are called inner B-splines. All other relevant B-splines are called outer B-splines.

- 30 The object of this invention is achieved by the process defined in claim 1. Special

embodiments of the invention are defined in the dependent claims. Claim 11 defines a computer system as claimed in the invention.

The right half of Figure 1 shows the incorporation of the process as claimed in the invention into the course of a FE simulation in the prior art and the substitution of certain process steps of a FE simulation in the prior art by the process of the present invention.

Input 1 of the simulation region Ω can be done via input devices, in particular also by storage of data derived from computer-aided engineering (CAD/CAM). For example, the data used in the engineering of a motor vehicle can be incorporated directly into the FE simulation as claimed in the invention.

In input 2 and storage of the type of boundary conditions natural and essential boundary conditions are distinguished. The basis as claimed in the invention is constructed for homogeneous boundary conditions of the same type. In particular, for essential boundary conditions, the basis functions vanish on the boundary Γ . Inhomogeneous boundary conditions can be treated in the assembly of the FE systems using methods which correspond to the prior art.

20

Finally, the control parameters are read in 3. They relate to the degree n and the grid width h of the B-splines to be used and the bound s for classification of the inner and outer B-splines. If specifications are omitted, all these input parameters can be automatically determined by evaluation of merit functions which are constructed empirically or analytically.

25

The following construction of the basis functions as claimed in the invention is divided into the steps shown schematically in Figure 3, which will now be described.

30

After reading in the simulation region Ω , in the first process step a grid covering the simulation region Ω is generated. Then it is checked which of the grid cells lie entirely inside, partially inside or not inside the simulation region Ω . The cell types 4 are determined, and this information about the cell types is stored. This
5 essentially requires inside/outside tests and determinations of intersections between the boundary Γ of the simulation region Ω and the segments or squares which bound the grid cells. Figure 7 shows the input and output data for this process step.

10 In the second process step, using the information about the cell types, the relevant B-splines are first determined. Then the classification 5 into outer B-splines is performed; the corresponding lists of indices are denoted by I and J . To this end, the size of those parts of the supports of the B-splines which lie within the simulation region is determined using the data obtained in the first process step,
15 and compared with the prescribed bound s . Figure 9 shows the input and output data for this process step.

In the third process step, coupling coefficients $e_{i,j}$ are computed 6; they join the inner and outer B-splines according to the rule

$$20 \quad B_i(x) = b_i(x) + \sum_{j \in J(i)} e_{i,j} b_j(x), \quad i \in I. \quad (1)$$

Hence, an extended B-spline B_i is assigned to each inner B-spline b_i . The construction and the properties of the index sets $J(i)$ and the coupling coefficients $e_{i,j}$ are given as follows. The index sets $J(i)$ consist of indices of outer B-splines. They correspond to complementary index sets $I(j)$ of indices of inner B-splines;
25 i.e., i belongs to $I(j)$ if and only if j belongs to $J(i)$. For a given outer index j the index set $I(j)$ is an array, i.e., a quadratic or cubical arrangement of $(n+1)^d$ inner indices which is characterized by a minimum distance to the index j . For a given outer index j and an inner index i in the index set $I(j)$, let p_i be the d -variate polynomial of degree n in each variable which has the value 1 at the point i and at

all other points of the array $I(j)$ the value 0. Then the coupling coefficient $e_{i,j}$ is given as the value of p_i at the point j , i.e., $e_{i,j} = p_i(j)$. The specific values of the coupling coefficient can either be tabulated for different degrees and relative positions of j and $I(j)$ or can be easily computed using Lagrange polynomials.

5 Figure 11 shows the input and output data for this process step.

If natural boundary conditions are given, the extended splines defined in equation (1) are used without modification for further implementation of the FE process. If conversely essential boundary conditions are given, a weighting according to the

10 rule

$$B_i(x) \leftarrow \frac{w(x)}{w(x_i)} B_i(x), \quad i \in I \quad (2)$$

has to be performed. The pertinent interrogation takes place in an optional process step 6a. The functions defined in this way are called weighted extended B-splines (WEB-splines). Formally, the extended B-splines used under natural boundary

15 conditions correspond to the special case $w(x) = 1$. They are, therefore, also called WEB-splines. For the case of essential boundary conditions, the weight function w is characterized as follows: For all points x of the simulation region, $w(x)$ can be bounded from above and below by positive constants, which are independent of x , times the distance $\text{dist}(x)$ of the point x from the boundary Γ . In other words, w
20 is positive within Ω and tends to zero in the vicinity of the boundary Γ as fast as the distance function dist . For simulation regions which are bounded by elementary geometrical objects (circles, planes, ellipses, etc.) a suitable weight function can optionally be given in explicit analytic form. Otherwise, computation rules should be used which typically represent a smoothing of the distance
25 function. The scaling factor $1/w(x_i)$ is calculated by evaluating the weight function at the weight point x_i . This can be any point in the support of the B-spline b_i which is at least half the bound $s/2$ away from the boundary.

As a result of the process as claimed in the invention, a computation rule for the
30 WEB-splines B_i (compare definitions (1) and (2)) is obtained which have all

favorable properties according to the object of the invention. Thus the FE method can be continued according to the prior art. But, in doing so, it is possible to advantageously exploit the regular grid structure of the basis functions according to claim 11.

5

In summary, it can be stated that for the process as claimed in the invention the coupling of the outer to the inner B-splines is, among other things, important. As a result, the constructed basis has the properties which are essential for FE computations. In particular, a basis B_i (i from the index set I) according to the
10 present invention, is uniformly stable with respect to the grid width h , and the error has the same order as for the B-splines b_k in the approximation of smooth functions which satisfy the same boundary conditions. On the other hand, the fulfilment of essential boundary conditions is ensured by using the weight function w .

15

The above described process steps and their properties as well as the preferred execution and the pertinent embodiments will be further explained and illustrated using the figures listed below. The features mentioned in the claims and in the specification can be essential to the invention individually or in any combination.

20 Figure 1 shows in the prior art the individual steps in the course of the finite element simulation and fits the determination of the WEB base into this course,

Figure 2a compares certain finite elements of the prior art to the WEB element shown in Figure 2b and lists the parameters relevant to
25 finite element approximations,

Figure 3 shows in a flow chart the process steps for determining the WEB basis,

Figures 4a and 4b show a support and the corresponding tensor product B-spline of degree 2,

Figures 5a and 5b illustrates the problem formulation of the first embodiment (displacement of a membrane under constant pressure) and shows the corresponding solution,

Figure 6 shows the cell types for the first embodiment,

5 Figure 7 surveys the input and output data of the process for determining the cell types,

Figure 8 illustrates, by way of example, the classification of the B-splines for the first embodiment,

Figure 9 outlines the input and output data of the process for classifying
10 the B-splines,

Figure 10 shows the coupling coefficients of an outer B-spline and the corresponding inner B-splines for the first embodiment,

Figure 11 surveys the input and output data of the process for computing the coupling coefficients,

15 Figures 12a and 12b illustrate the construction of the weight function of the preferred embodiment,

Figure 13 shows the support of a WEB-spline and the corresponding coupling coefficients for the first embodiment,

Figures 14a and 14b explain the problem formulation of a second
20 embodiment (incompressible flow) and its solution using the flow lines and a figure for the distribution of the flow velocity,

Figures 15a to 15c show the changing classification of B-splines for the B-spline degrees $n = 1, 2, 3$ and the same grid width for the second embodiment,

25 Figures 16a to 16c provide information about the error development in the finite element approximation using WEB-splines and about the computing time behavior of the WEB approximations for the second embodiment,

Figures 17a and 17b compare the WEB basis to a process based on linear trial functions on a triangulation (prior art), and

Figure 18 shows a computer system as claimed in the invention.

One especially favorable embodiment of the process as claimed in the invention,
5 called the WEB process, is determined by the following specifications.

The bound s is chosen such that the inner B-splines are characterized by requiring that at least one of the grid cells of their support lies completely in the simulation region Ω . Since to determine the relevant B-splines, the intersection of the grid
10 cells and the boundary Γ must be computed anyway, the classification requires no significant additional computing time. The weight point x_i is chosen as the midpoint of a grid cell in the support of the B-spline b_i which lies completely in the simulation region Ω . This is also efficiently possible since the determination of one such cell is already part of the classification routine.

15

If no explicit analytic form of the weight function is known, it is defined by

$$w(x) = \begin{cases} 1 & \text{if } \text{dist}(x) \geq \delta \\ 1 - (1 - \text{dist}(x)/\delta)^n & \text{if } \text{dist}(x) < \delta. \end{cases} \quad (3)$$

Figures 12a and 12b illustrate the construction of the weight function. Here, the parameter δ indicates the width of the strip Ω_δ within which the weight function
20 varies between the value 0 on the boundary of the simulation region and the value 1 on the plateau on $\Omega \setminus \Omega_\delta$. The parameter δ is chosen such that the smoothness of the weight function is ensured.

One important advantage of the process is that no meshing of the simulation
25 region is necessary. In technical applications, this results in clear savings of computing time and storage capacity and simplifies the course of the simulation. The process structure for two- and three-dimensional problems is formally and technically largely identical. This enables time- and cost-saving implementations of

solvers for diverse applications based on uniform program structures. The use of B-splines corresponds to the industrial standard in the modeling of geometrical objects, and thus forms a natural connection between FE and CAD/CAM applications. Extensive existing program libraries from both fields can be used for
5 implementing a FE simulation based on the process as claimed in the invention. The basis functions constructed using the WEB process have all standard properties of finite elements. This includes especially the stability of the basis. It implies, for example, that for linear elliptic boundary value problems the condition number of the resulting system of equations does not grow faster than for optimal
10 triangulations as the grid width becomes smaller. For applications, this means, for example, that linear systems of equations as they typically arise in FE methods can be efficiently solved by iterative algorithms. Furthermore, for a given degree, the approximation order is maximal and the number of necessary parameters minimal. Thus, very accurate approximations are possible with a relatively small
15 number of parameters. Specifically, this can mean that the accuracies which so far required the use of mainframe computers can now be achieved with workstations. The regular grid structure of the basis as claimed in the invention permits a very efficient implementation, especially for assembling and solving FE systems. Moreover, by using the weight function, the boundary conditions can be satisfied
20 during simulation without affecting the regular grid structure of the basis functions. Finally, by using multigrid methods to solve the linear systems arising in linear elliptic boundary value problems, one can achieve that the overall solution time is proportional to the number of unknown coefficients, and thus optimal.

25 The process as claimed in the invention in the special preferred embodiment (WEB process) is illustrated using the first embodiment shown in Figures 5a and 5b. The differential equation and boundary conditions are chosen to be very elementary so that in addition to the construction of the WEB basis as claimed in the invention the entire course of the FE simulation can be followed without major
30 additional effort.

Figure 5a shows an elastic membrane which is fixed along the boundary Γ of a planar simulation region Ω , and on which a constant pressure $f = 1$ acts inside the region. With suitable normalization, the displacement u satisfies the Poisson

5 equation with homogeneous boundary conditions,

$$\begin{aligned} -\Delta u &= 1 \quad \text{in } \Omega \\ u &= 0 \quad \text{on } \Gamma. \end{aligned}$$

The displacement u or the deflection of the membrane is depicted in Figure 5b. As
10 described above, the WEB process is divided into the following steps.

Input 1 of the simulation region Ω : The boundary Γ is a periodic spline curve of degree 6, which is stored by its control points 20 (in Figure 5a identified with black dots).

15

Input 2 of the boundary conditions: the homogeneous boundary condition is essential so that the construction of a weight function is necessary.

Input 3 of the control parameters: The degree $n = 2$, and, in order to make the
20 figures easier to understand, a relatively large grid width $h = 1/3$ are used.

Determination 4 of the cell types: As illustrated in Figure 6, the simulation region is covered by a grid 21, which contains the grid cells of the supports of all B-splines of potential relevance for the basis construction. The type determination
25 in the example yields 69 outer grid cells 22, and 11 inner grid cells 24, and 20 grid cells 23 on the boundary.

Classification 5 of the B-splines: Here, the support of the B-spline b_k is a square Q_k with corners

$$(k_1, k_2)h, (k_1 + 3, k_2)h, (k_1 + 3, k_2 + 3)h, (k_1, k_2 + 3)h;$$

$Q_{(-4,0)}$ and $Q_{(2,1)}$ are shown in Figure 8. The grid points kh of the relevant B-splines, for which Q_k intersects the interior of the simulation region, are marked in Figure 8 by a point or a circle. All grid points ih for inner B-splines (i from the index list I), for which at least one cell of the support Q_i lies entirely within Ω , are marked by a point. For example, for $i = (-4, 0)$, the grid cell $(-2, 0)h + [0, h]^2$ lies entirely in Ω . All grid points jh for outer B-splines (j from the index list J), for which no cell of the support Q_j lies entirely in Ω , are marked by a circle.

Computation 6 of the coupling coefficients: To determine the coupling coefficients $e_{i,j}$ for each fixed j of the index list J the nearest 3×3 -array

$$I(j) = \{\ell_1, \ell_1 + 1, \ell_1 + 2\} \times \{\ell_2, \ell_2 + 1, \ell_2 + 2\}$$

of indices in I is sought. In Figure 10, for the outer grid point $j = (-1, 2)$, which is marked with a circle, the array $I(j)$ is identified with points. Figure 10 likewise shows the corresponding coupling coefficients in a matrix representation. They are computed by bivariate interpolation. For example, the interpolating polynomial for $i = (-1, -1)$ is

$$p_i(x) = -x_1(x_1 + 2)x_2(x_2 - 1)/2.$$

Its value at the point $x = j = (-1, 2)$ is $p_i(j) = 1 = e_{i,j}$. One notices that many of the coupling coefficients are 0. This is a typical phenomenon. The coupling coefficients $e_{i,j}$ are not equal to 0 for all i of the index list $I(j)$ only if the indices i are different from the index j in each component.

Computation rule 7 for the weight function: The weight function is given by equation (3) with $n = 2$ and $\delta = 0.2$. The parameter δ is computed numerically. It must be small enough to avoid singularities of the distance function. To compute the distance function, the process generates a program which uses Newton's method. Since the weight function is not equal to 1 only in a boundary strip, the complexity in the subsequent evaluation is low.

Output: Figure 13 shows the support of a WEB-spline B_i and the data necessary for its description. These are the index list $J(i)$ of the outer B-splines b_j coupled with b_i , the coupling coefficients $e_{i,j}$, and the weight point x_i . These data are used in conjunction with the weight function for generating the computation rule for
5 the WEB-splines.

The further course of the FE simulation follows the prior art.

Assembly 9 of the FE system: The entries of the system matrix and of the
10 right-hand side are

$$G_{k,i} = \int_{\Omega} \text{grad } B_k \text{ grad } B_i, \quad F_k = \int_{\Omega} f B_k, \quad k, i \in I.$$

The system of equations $GC = F$ for the basis coefficients C_i in this example has dimension 31. The matrix entries $G_{k,i}$ are computed using numerical integration, likewise the integrals F_k .

15

Solution 10 of the FE system: The Galerkin system is solved iteratively with the conjugate gradient method with SSOR preconditioning used to accelerate convergence. After 24 iteration steps the solution is found within machine accuracy (tolerance $\leq 1e - 14$).

20

Computation 11 and output 12 of the approximation: The approximation computed with the process as claimed in the invention is $u = \sum_i C_i B_i$ and is shown graphically in Figure 5b. The relative error of the L_2 -norm is 0.028.

25 The efficiency of the process as claimed in the invention in the special preferred embodiment (WEB process) is illustrated in a second embodiment using the simulation of an incompressible flow. The arrangement of two circular obstacles shown in Figure 14a in a channel with parallel boundaries serves to illustrate the principal strategy. For complicated geometries, as are typically present in specific
30 applications, the process works completely analogously and efficiently. In

Figure 14a the stream lines 25 are shown within the region bounded by Γ_1 to Γ_4 and by Γ_5 and Γ_6 . The differential equation is:

$$\Delta u = 0 \quad \text{in } \Omega$$

with the boundary conditions

$$5 \quad \frac{\partial u}{\partial n} = v_0 \text{ on } \Gamma_1, \quad \frac{\partial u}{\partial n} = -v_0 \text{ on } \Gamma_2, \quad \frac{\partial u}{\partial n} = 0 \text{ on } \Gamma_3, \dots, \Gamma_6.$$

The flow velocity $v = -\text{grad } u$ is shown in Figure 14b.

The construction of the WEB basis as claimed in the invention proceeds completely analogously to the first embodiment. The sole difference is that a
10 weight function is not necessary because of the natural boundary conditions.

Figures 15a to 15c show the classification of the relevant B-splines for different degrees n (see also Figure 8). In the figure, the inner B-splines b_i , which are taken into the WEB basis without extension, are marked by solid triangles. For small h
15 the number of these B-splines increases, i.e., $B_i = b_i$ for most of the WEB basis. For degree $n = 3$ this is the case for 236 of 252 indices i in the example.

Figure 16a shows in two diagrams the numerically determined relative L_2 -error of the potential (left half of the figure) as a function of the grid width $h = 2^{-k}$ with
20 $k = 1, \dots, 5$ and the numerically estimated order of convergence m (right half of the figure). Here, for different degrees of the WEB-spline the following markers are used: $*$ ($n = 1$), \circ ($n = 2$), \triangle ($n = 3$), \square ($n = 4$) and \star ($n = 5$). As expected, $m \approx n + 1$, i.e., an approximate error reduction by a factor 2^{n+1} when the grid width is cut in half. Analogously, for the relative approximation error of the flow
25 velocity shown in Figure 16b (H^1 -norm of the solution, left half of the figure), an order of convergence $m \approx n$ (right half of figure) is obtained with an associated error reduction by roughly a factor 2^n when the grid width is cut in half.

Figure 16c (right half of the figure) shows the computing time in seconds for construction of the WEB basis as a function of the number of resulting basis functions, measured on a Pentium II processor with 400 MHz. For example, for construction of a WEB basis of degree 3 with grid width $h = 0.125$ with 2726
5 WEB-splines 1.32 seconds are necessary. One notices that the complexity for generating the WEB basis is largely independent of the degree n of the basis. In the left half of Figure 16c the number of CG-iterations relative to the number of basis functions is shown. Thus, for the corresponding system with 2726 unknowns, 65 PCG-iterations are required. The total computing time including assembling
10 and solving the Galerkin system is roughly 2.48 seconds.

Figures 17a and 17b compare the WEB process with a standard solution process which meshes or triangulates the simulation region (Fig. 17a) and uses hat
functions. The graph shows in Fig. 17b the L_2 -error relative to the number of
15 parameters. The results of the standard solver are marked with boldfaced diamonds and are compared to the results achieved using the WEB basis of degrees 1 to 5. For example, an accuracy of 10^{-2} is achieved with the WEB process by using 213 basis functions with degree 2 and an overall computer time of 0.6 seconds. To achieve the same accuracy, the standard method with linear hat
20 functions required 6657 basis functions.

In the assessment of the standard solution process two other aspects must be considered. On the one hand, Figure 17b illustrates that even a moderate accuracy of 10^{-3} can only be achieved with hat functions when far more than one million
25 coefficients are used. This shows that when using hat functions accurate results generally require an enormous computing and storage capacity or cannot be achieved at all with the prior art. On the other hand, the complexity required for meshing increases with the complexity of the simulation region. In contrast to realistic applications, the region studied here is comparatively simple to
30 triangulate due to its simple structure.

The two-dimensional example shows the performance gain by the WEB process. An even greater increase in performance is possible in three-dimensional problems. On the one hand, the complexity for meshing, which is eliminated in the WEB
5 process, is much greater. On the other hand, the reduction in the number of required basis functions becomes much more noticeable than in the two-dimensional case.

Figure 18 shows a device as claimed in the invention, especially a computer system
10 30, with input devices 31, 32, 33, output devices 34 and a control unit 35 which controls the course of the process. To carry out the process as claimed in the invention and in particular for purposes of parallelization of the pertinent computations, the central control unit 35 preferably uses several arithmetic logic units (ALU) or even several central processing units (CPU) 36. These allow
15 especially parallel processing for the process steps classification 5 of the B-splines, in particular also intersection of the regular grid with the simulation region Ω , determination 6 of the coupling coefficients $e_{i,j}$, and/or evaluation of the weight function $w(x)$ at points x of the simulation region Ω .

20 The computer units 36 here access the common data resources of the storage units 37. The data can be input, for example, by a keyboard 31, a machine-readable data medium 38 via a corresponding read station 32 and/or via a wire based or wireless data network with a receiver station 33. Via the read station 32 or a pertinent data medium 38, the control program, which controls the process
25 execution, can be input, and, for example, can be permanently filed on the storage media 37. Accordingly, the output devices 34 can be a printer, a monitor, a write station for a machine-readable data medium and/or the transmitting station of a wire based or wireless data network.

Claims

1. A process for increasing the efficiency of a computer system for finite element simulations by automatic generation of suitable basis functions using B-splines, with the following steps:
 - 5 • definition (1) of a simulation region (Ω) and storage of the data of the simulation region (Ω),
 - input (2) and storage of boundary conditions,
 - establishment (3) of a predefinable grid width h and a predefinable degree n of the B-splines,
 - 10 • determination (4) of a grid covering the simulation region (Ω) and the type of the grid cells,
 - classification (5) of the B-splines with support intersecting the simulation region (Ω) by determining inner and outer B-splines, where for outer B-splines the intersection of the support with the simulation
 - 15 region (Ω) is less than a predefinable bound s ,
 - determination (6) of coupling coefficients $e_{i,j}$ for formation of linear combinations of inner and outer B-splines and
 - storage and output of the parameters which determine the basis functions.
- 20 2. Process as claimed in claim 1, wherein, before storage and output of the parameters, the following step is carried out: Establishing (7) a predefinable weight function w and determining (8) the weight points and scaling factors.
3. Process as claimed in claim 2, wherein the weight function w is established by a smooth transition from a constant plateau inside the simulation region
- 25 (Ω) to the value zero on the boundary (Γ).

4. Process as claimed in one of the claims 1 to 3, wherein the B-splines with at least one grid cell of the support contained entirely in the simulation region (Ω) are classified as inner B-splines.
5. Process as claimed in one of the claims 1 to 4, wherein the weight point is
5 chosen as the midpoint of a grid cell of the support of the corresponding B-spline which is contained entirely in the simulation region (Ω).
6. Process as claimed in one of the claims 1 to 5, wherein the simulation region (Ω) is defined by storage of data which can be derived from computer-aided engineering (CAD/CAM).
- 10 7. Process as claimed in one of the claims 1 to 6, wherein the grid width h is automatically established using stored values obtained empirically and/or analytically by a pertinent first merit function.
8. Process as claimed in one of the claims 1 to 7, wherein the degree n is automatically determined using stored values obtained empirically and/or
15 analytically by a pertinent second merit function.
9. Process as claimed in one of the claims 1 to 8, characterized by the following steps:
 - assembling (9) a system of equations to be solved in the FE simulation,
 - solving (10) the system of equations,
 - 20 • computing (11) an approximate solution, and
 - outputting (12) of the approximate solution.
10. Process as claimed in claim 9, wherein a multigrid process is used for the solution (10) of the system of equations.
11. Device for executing a process as claimed in one of the claims 1 to 10, in
25 particular a computer system, with input devices (31,32,33) and output devices (34), storage devices (37), and a central processing unit (35,36),

where the regular grid structure is utilized for optimizing the computational process, especially by parallelization.

12. Machine-readable data medium (38), in particular magnetic tape, magnetic disk, compact disk (CD) or digital versatile disk (DVD), wherein the data
5 medium (38) stores a control program for a computer system (30), according to which the computer system (30) can execute a process, as claimed in one of the claims 1 to 10.

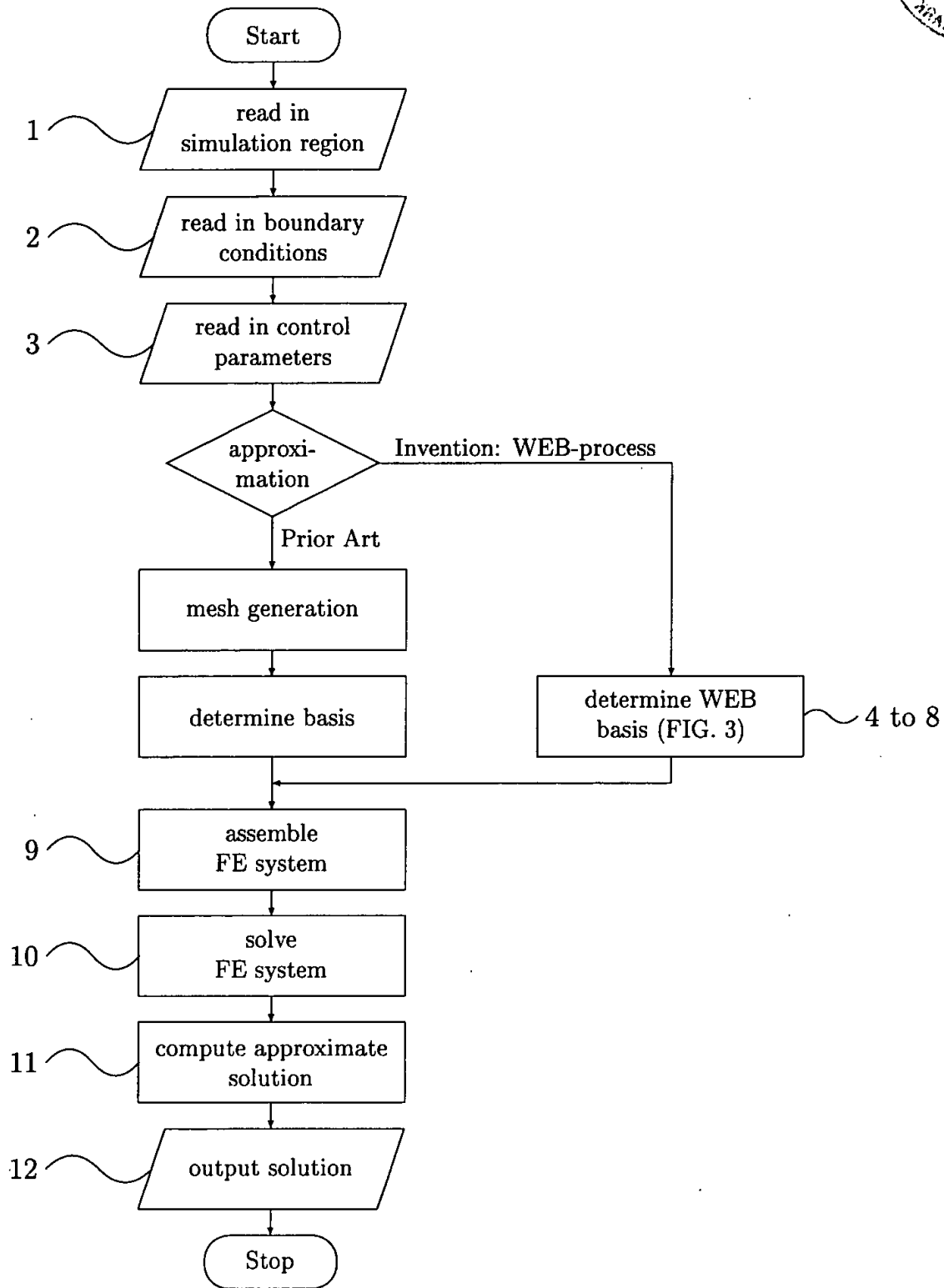
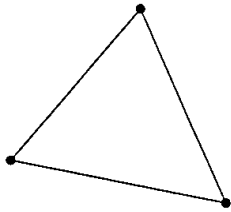


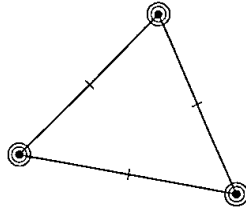
Fig. 1



Linear Element



Argyris-Triangle



Clough-Tocher Element

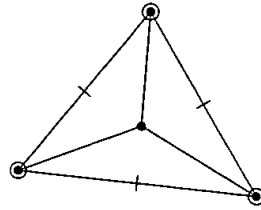
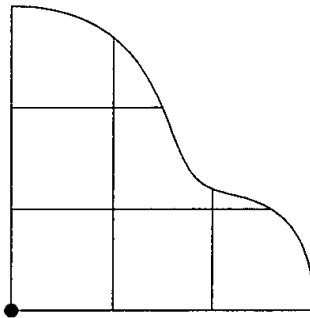


Fig. 2a
Prior Art

WEB-Spline



Type	Dimension	Degree	Smoothness	Parameters
linear element	d	1	0	$d + 1$
Argyris triangle	2	5	1	21
Clough-Tocher element	2	3	1	12
WEB-Spline	d	n	$n - 1$	1

Fig. 2b

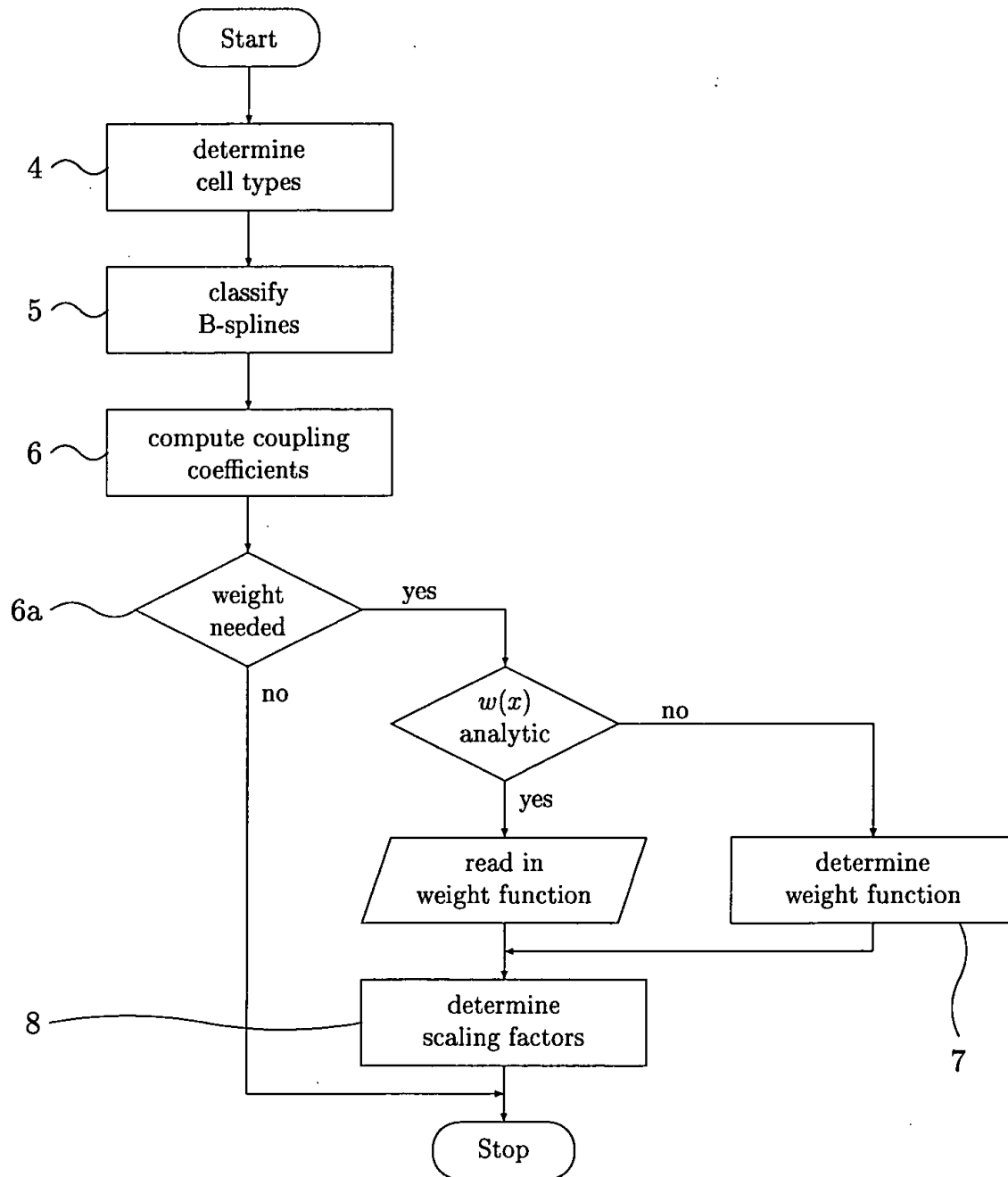
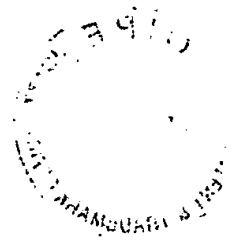


Fig. 3

0122 3910
 1957 JAN 16 PM 4 1921

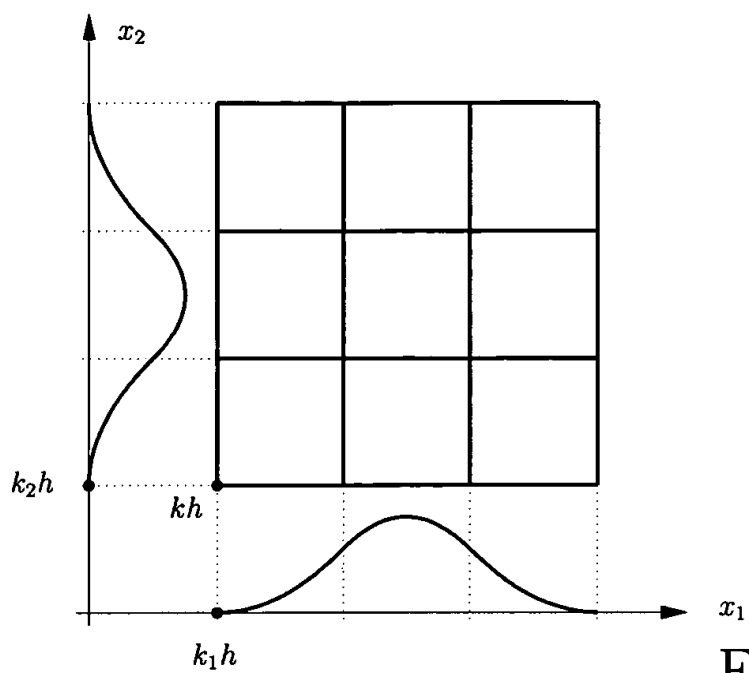


Fig. 4a

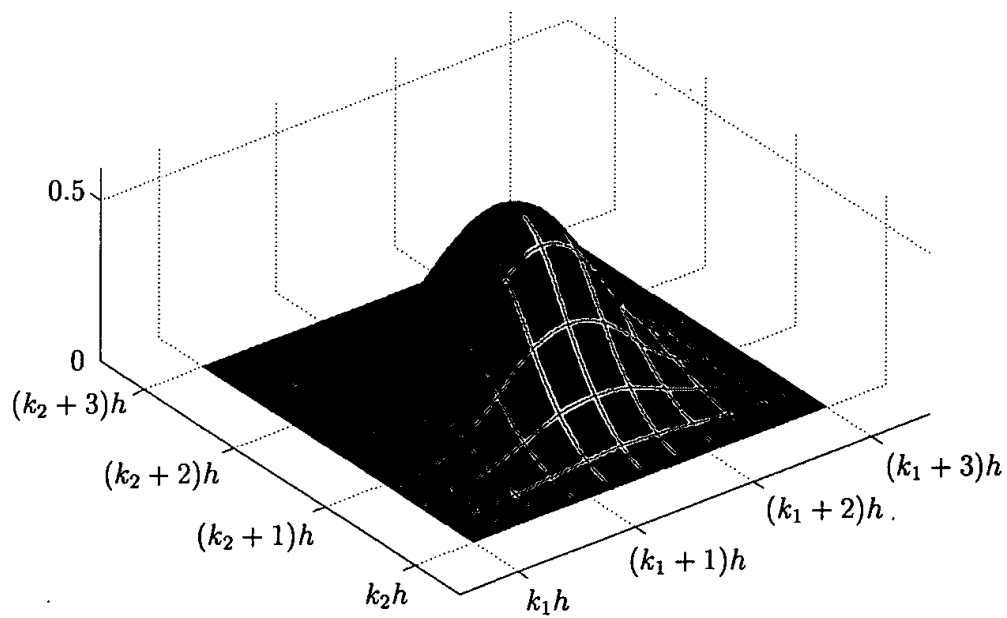


Fig. 4b

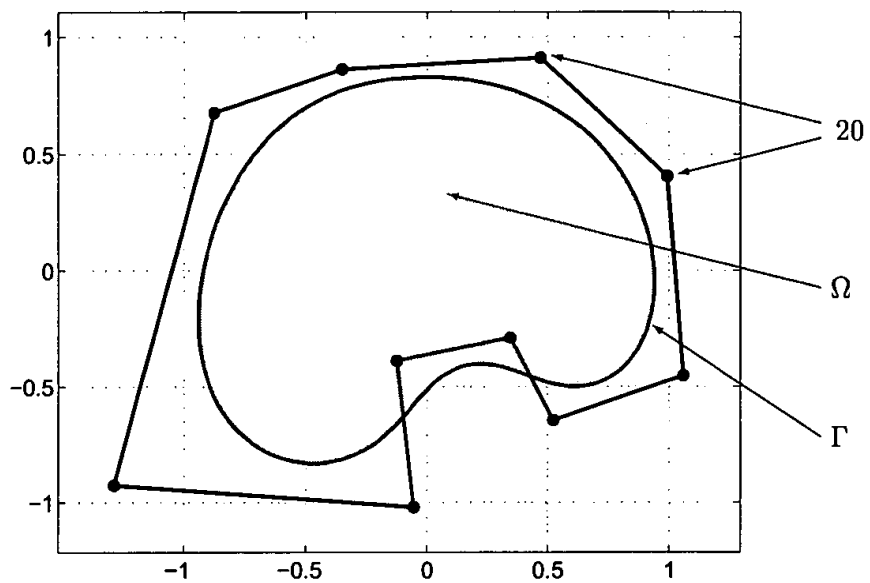


Fig. 5a

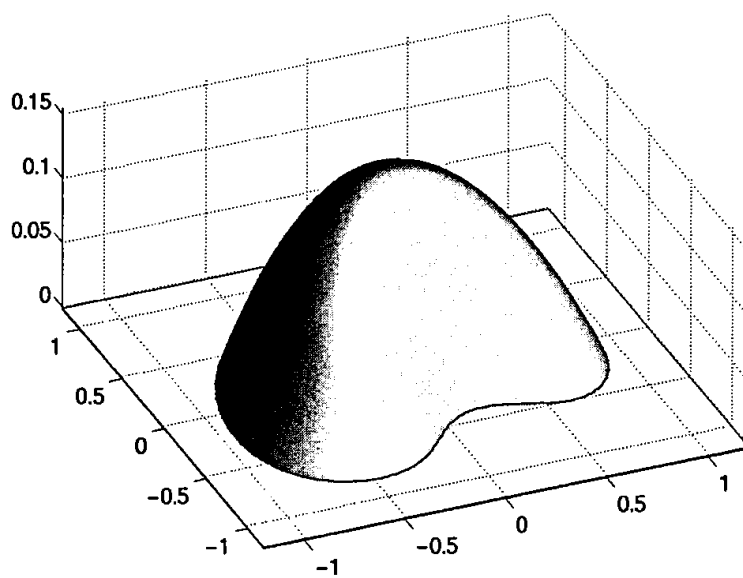


Fig. 5b

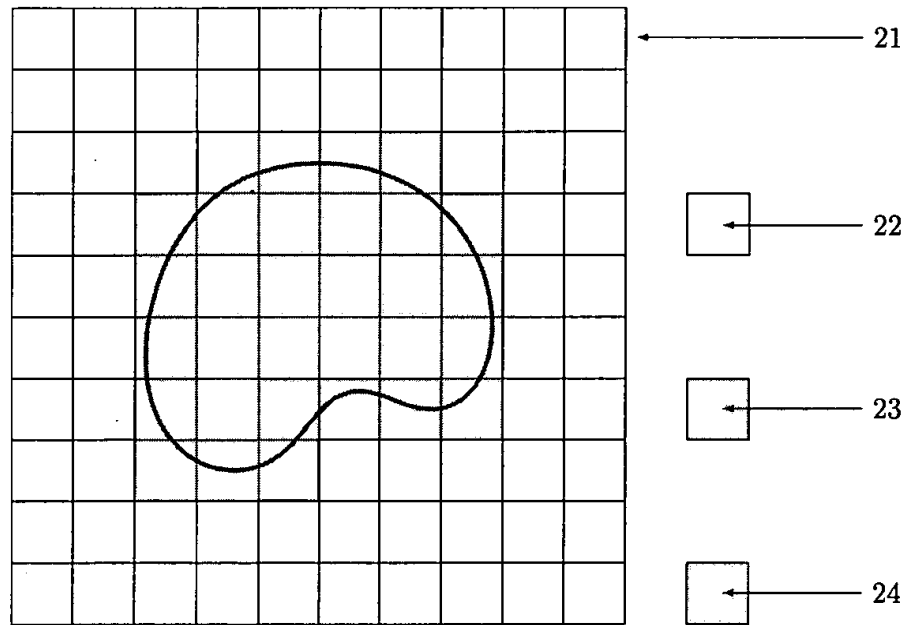


Fig. 6

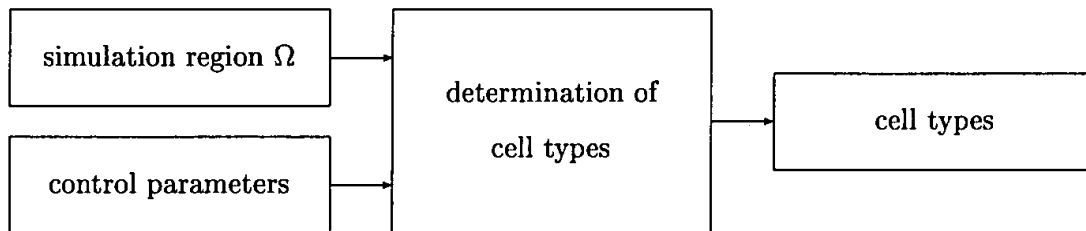


Fig. 7

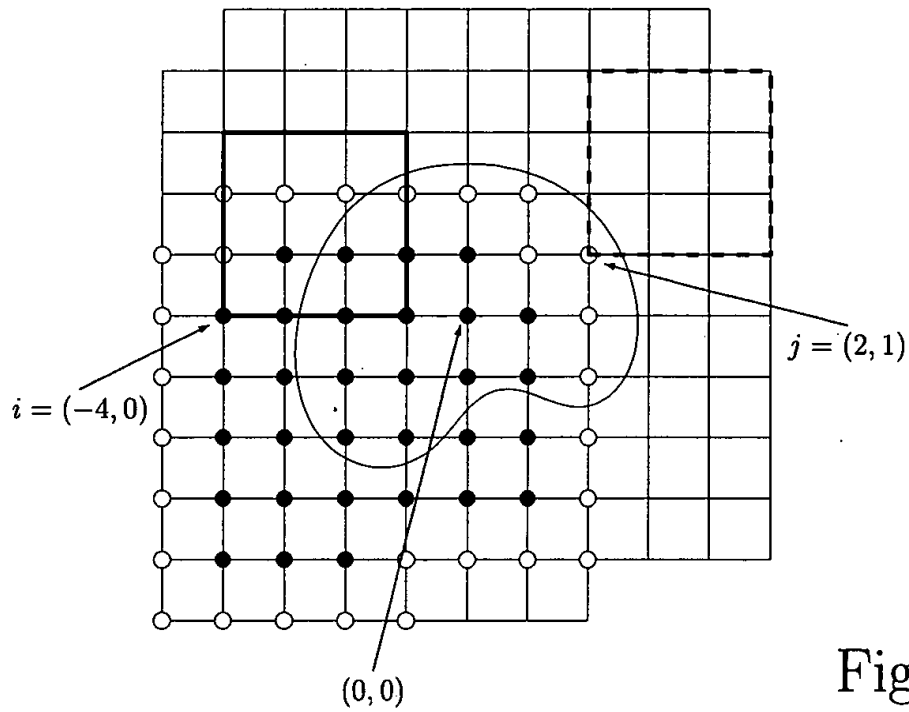


Fig. 8

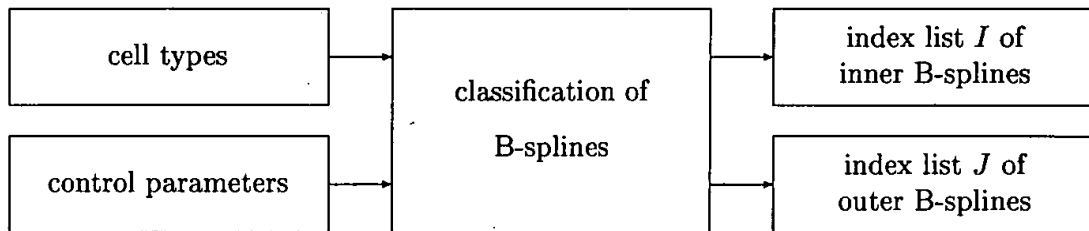


Fig. 9

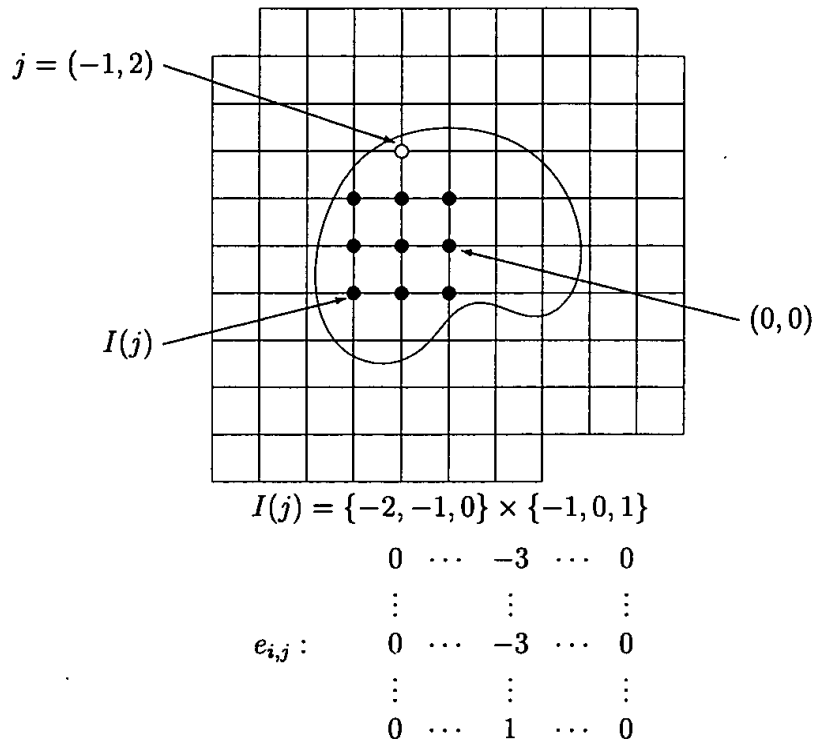


Fig. 10

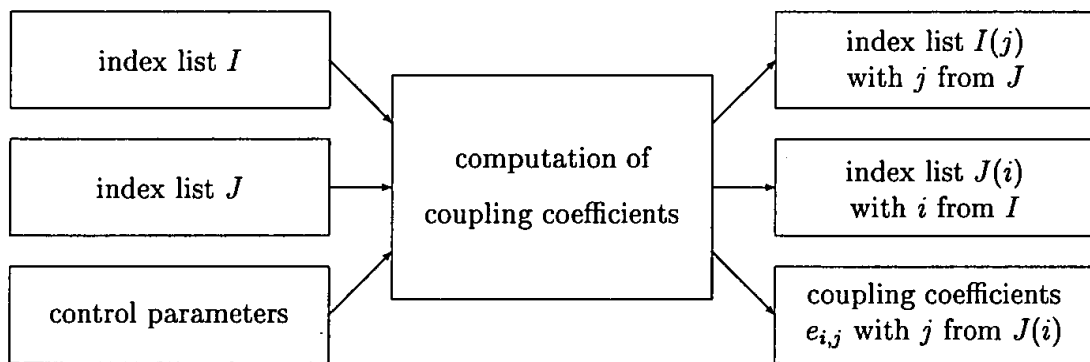


Fig. 11

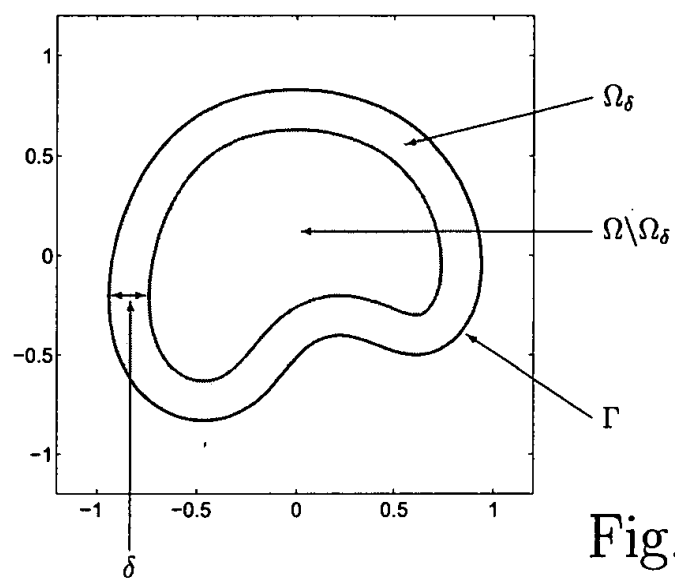


Fig. 12a

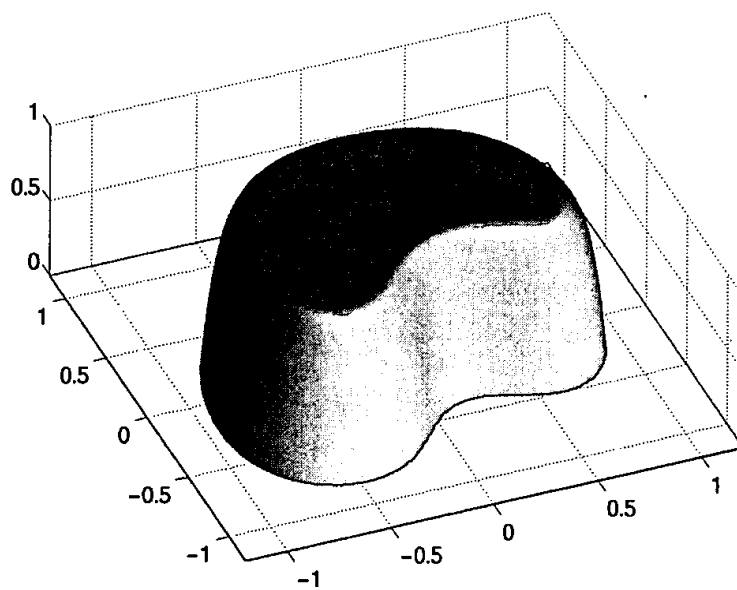
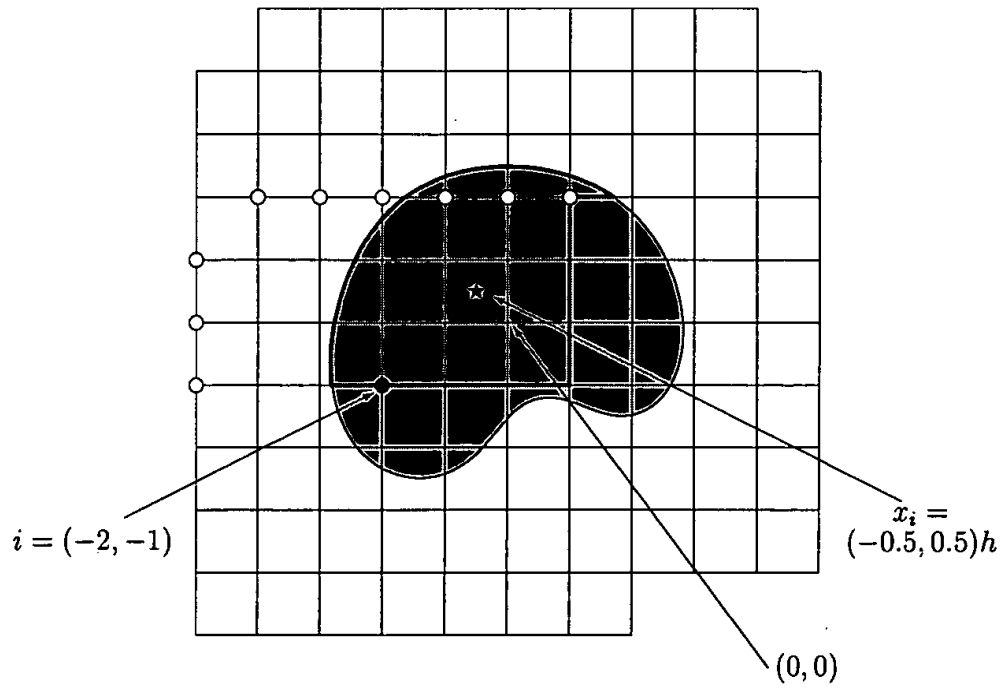


Fig. 12b



$$J(i) = \{(-5, -1), \dots, (-5, 1), (-4, 2), \dots, (1, 2)\}$$

$$e_{i,j} : \begin{matrix} \cdot & \dots & -3 & \dots & 0 & \dots & 1 & \dots & 0 & \dots & 0 & \dots & 1 \\ \vdots & & & & & & & & & & & & \\ -3 & & & & & & & & & & & & \\ \vdots & & & & & & & & & & & & \\ 0 & & & & & & & & & & & & \\ \vdots & & & & & & & & & & & & \\ 1 & & & & & & & & & & & & \end{matrix}$$

Fig. 13

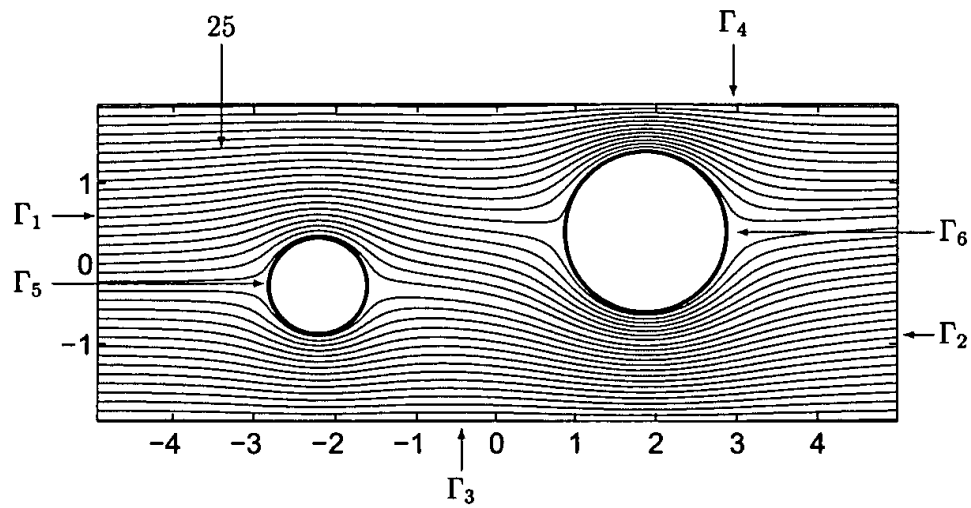


Fig. 14a

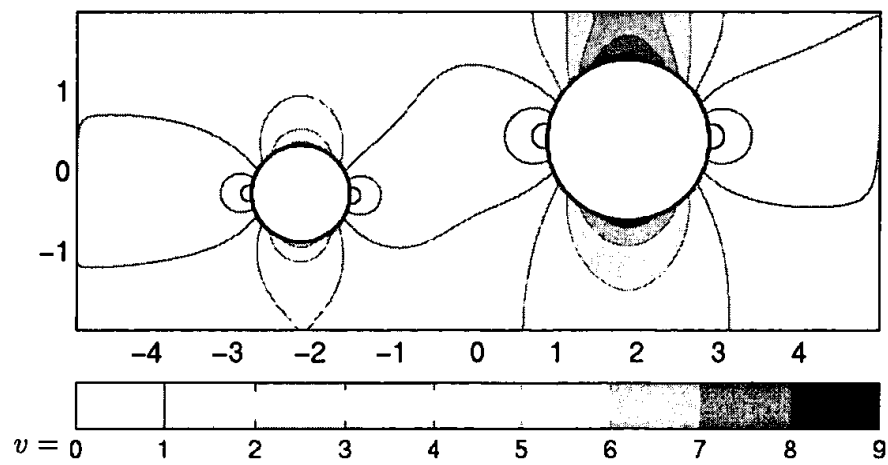


Fig. 14b

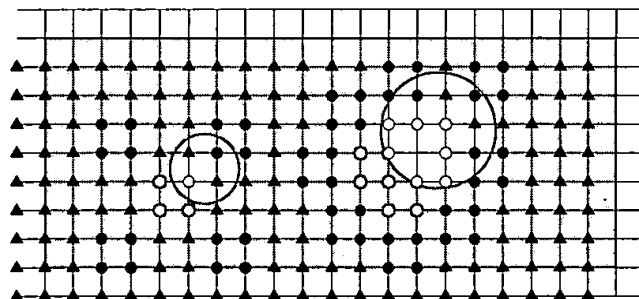


Fig. 15a

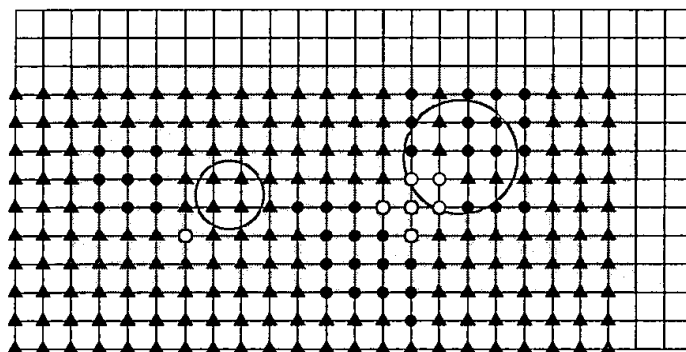


Fig. 15b

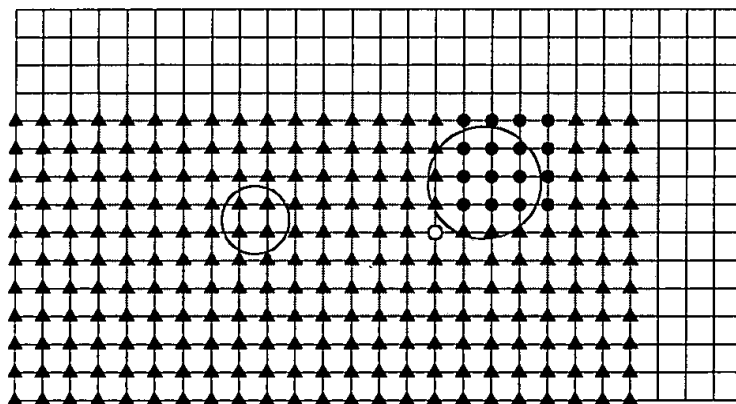


Fig. 15c

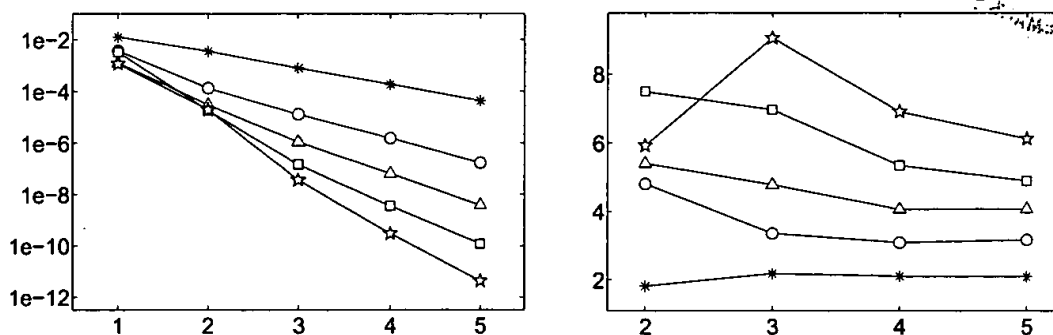


Fig. 16a

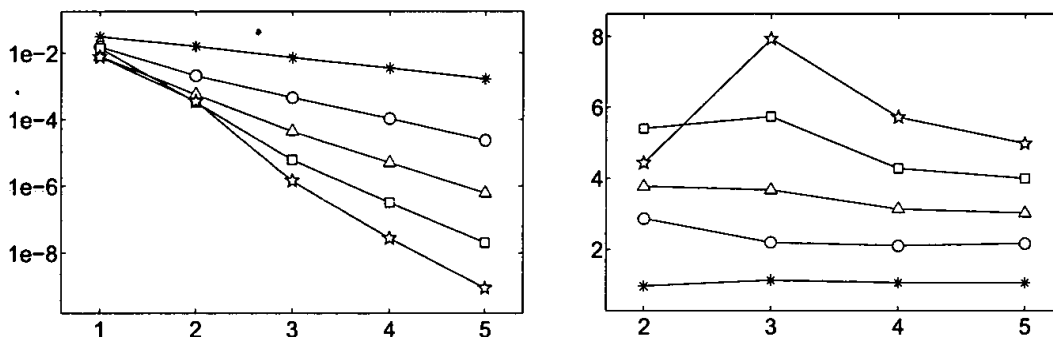


Fig. 16b

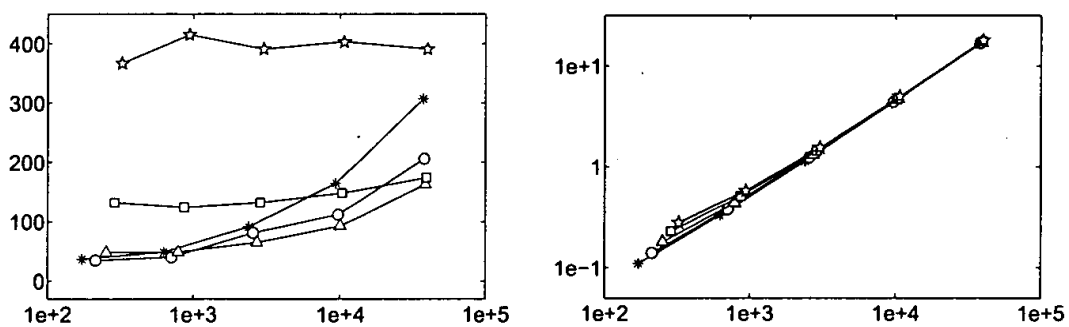


Fig. 16c

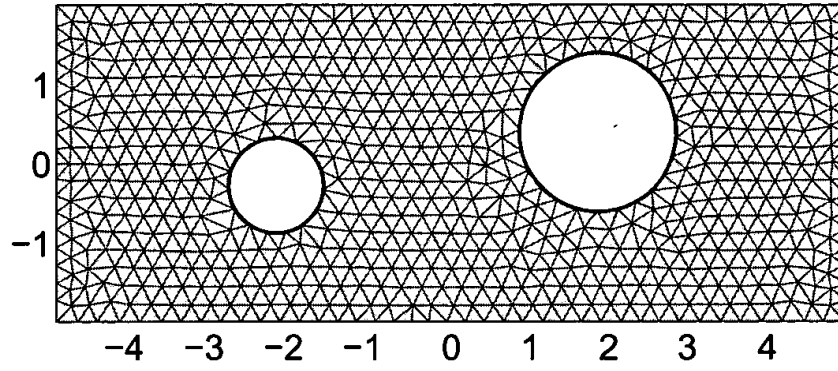


Fig. 17a

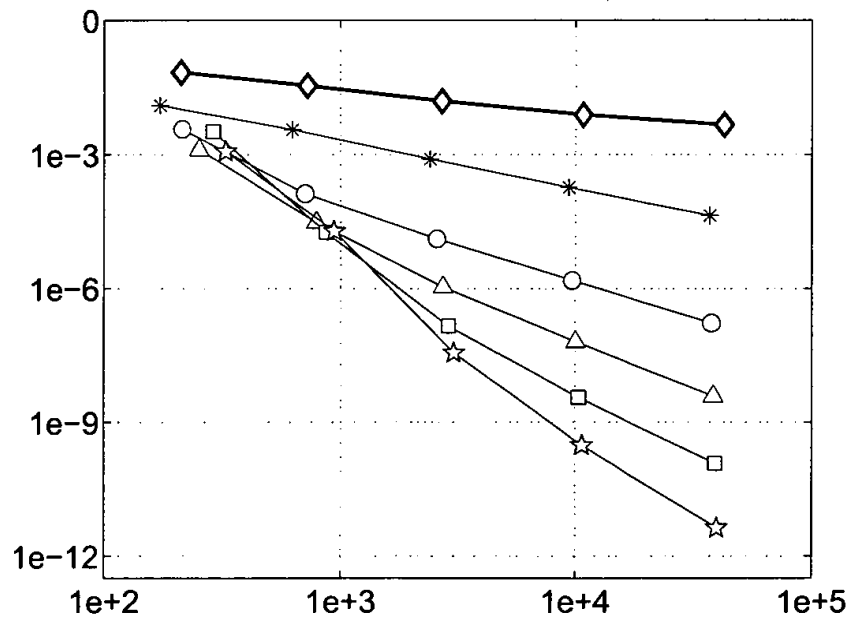


Fig. 17b

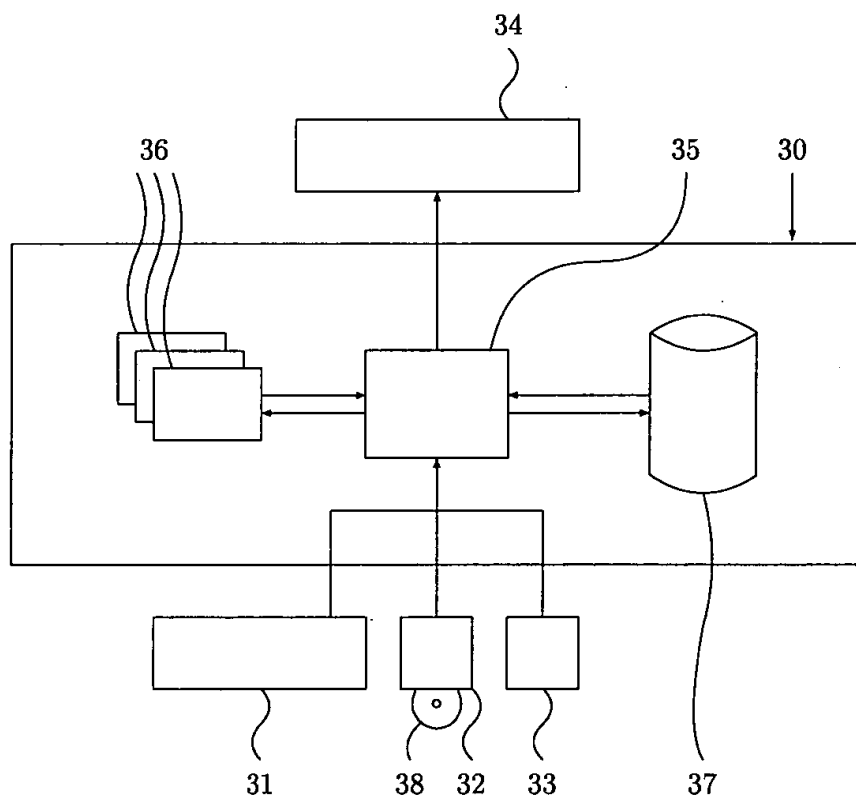


Fig. 18

Abstract

The invention relates to a process for increasing the efficiency of a computer system in finite element simulations by efficient automatic construction of suitable basis functions for computing approximate solutions and such a computer system.

- 5 In the process as claimed in the invention, a grid covering the simulation region Ω is generated. B-splines defined thereon with supports which intersect the simulation region are classified into inner and outer B-splines (5). Then, coupling coefficients for forming linear combinations of inner and outer B-splines are determined (6), and the parameters which determine the resulting basis functions,
10 are stored and output.

Figure 3

List of designations and abbreviations

B_i	(weighted) extended B-splines (WEB-splines)
b_i	inner B-splines
b_j	outer B-splines
b_k	relevant B-splines
d	dimension of B-Splines
dist	distance function
dist(x)	distance of point x from boundary Γ
$e_{i,j}$	coupling coefficients
f	perturbation function (right hand side of the differential equation)
FE	Finite Element
h	grid width, edge length
I	index set of the inner splines
$I(j)$	index set of the inner splines coupled to an outer spline
i	index of an inner spline
J	index set of the outer splines
$J(i)$	index set of the outer splines coupled to an inner spline
j	index of an outer spline
k	d -dimensional grid index
m	order of convergence
n	degree of B-splines
p_i	d -variate polynomial of degree n
Q_k	support of B-spline with index k
s	bound of the support portion in Ω
u	solution of the differential equation
v	flow velocity
$w(x)$	weight function

WEB	weighted extended B-spline
x_i	weight point in the simulation region
Z_k	grid cells
δ	parameter, the width of the strip in which the weight function rises
Γ	boundary of the simulation region
Ω	simulation region
1	definition of the simulation region
2	input and storage of boundary conditions
3	establishment of control parameters
4	determination of a grid and cell classification
5	classification of the B-splines
6	determination of the coupling coefficients
7	determination of a weight function
8	determination of weight points and scaling factors
9	assembling of a system of equations
10	solution of the system of equations
11	computation of an approximate solution
12	output of the approximate solution
20	control points
21	grid
22	outer grid cells
23	grid cells on the boundary
24	inner grid cells
25	stream lines
30	computer system

- 31 keyboard
- 32 read station
- 33 receiving station
- 34 output devices
- 35 central control unit
- 36 central processing unit
- 37 storage device
- 38 data medium

Erklärung zur Übersetzung:

Diese Übersetzung ist eine überarbeitete Fassung der von Lawrence B. Hanlon vom International Translation Center Inc. erstellten literalen Übersetzung.

Korrigiert wurden neben den Fachbegriffen und fachspezifischen Formulierungen lediglich einige fehlerhaft übersetzte Sätze.

Aufbauend auf der Erklärung des Übersetzers Lawrence B. Hamilton erklären die Autoren Klaus Höllig, Ulrich Reif und Joachim Wipper hiermit, dass nach bestem Wissen und Gewissen die literale Übersetzung mit dem deutschen Original übereinstimmt.

Technical Terms to be Checked

German	English
Ausgabemittel	output device
Bewertungsfunktion	merit function
Computereinrichtung	computer system
Datenträger	data medium
Eingabemittel	input device
Empfangsstation	receiving station
Feld (quadratisch/würfelförmig)	array
Lesestation	read station
Recheneinheit	computer unit
Speichermittel	storage means
Steuereinrichtung	control unit
Zentrale Steuereinrichtung	central control unit

Technical Terms

German	English
Basis	basis
Basisfunktion	basis function
Feld (von Indizes)	array of indices
Gewichtsfunktion w	weight function w
Gitter	grid
Gitterweite	mesh width
Näherungslösung	approximate solution
Netz	mesh
Netzgenerierung	mesh generation
Vernetzung	meshing
Rechenzeit	computing time
Träger	support
verschwinden (auf Γ)	vanish (on Γ)